

Alles über Grafik

Ausführliche Kurse

- Grafik perfekt programmiert
- Zaubern mit Shapes und Sprites

Das Super-Tool

- Power 128:
  Directory komfortabel organisieren

Anspruchsvolle Anwendungen

- Räumliche Grafik:
  3D-Landschaften aus dem Computer
- Schachverwaltung:
  Professionelle Turnierauswertung
- Haushaltsbuch:
  Finanzen endlich im Griff

Tolle Spiele zum Abtippen

Alle Programme auch auf Diskette erhältlich



Exklusiv bei Markt&Technik

F. Müller/T. Petrowski
C64 - GEOS 1.3 deutsch
1988, 571 Seiten, inkl. Diskette
Brandaktuell: das Anwendungs-, Programmier- und Systemhandbuch jetzt für die neueste GEOS-Version 1.3.
Bestell-Nr. 90570, ISBN 3-89090-570-6
DM 59,- sFr 54,30/öS 460,20

F. Müller/T. Petrowski
C64 - Alles über GEOS 1.2
1987, 461 Seiten, inkl. Diskette
Das umfassende Buch über Anwendung und Programmierung der grafischen Benutzeroberfläche GEOS mit vielen Hilfs- und Beispielprogrammen auf Diskette. Die Diskette enthält auch einen Makroassembler und Monitor zur Programmierung, Manipulation und Analyse von GEOS.
Bestell-Nr. 90461, ISBN 3-89090-461-0
DM 49,- sFr 45,10/öS 382,20







F. Müller
C64 für Insider
1988, 550 Seiten
Das ROM-Listing ist eines der wichtigsten Werkzeuge des ernsthaften Programmierers. Ebenso benötigt er eine komplette Beschreibung der RAM-Speicherzellen, die von Betriebssystem und Basic-Interpreter verwendet werden.
Die Besonderheit dieses Werkes liegt in der einzigartigen Aufbereitung der ROM-Routinen, der dokumentierten Speicherbelegung (Memory Map) und einem Begleittext, der die internen Zusammenhänge selbst einem Anfänger verständlich macht. Auch wenn Sie schon ein C64-ROM-Listing besitzen – dieses neue Werk sollten Sie sich nicht entgehen lassen. Selbst ausgefuchsten Profis bietet es noch neue Informationen.
Bestell-Nr. 90481, ISBN 3-89090-481-5
DM 59,- sFr 54,30/öS 460,20



















W. Besenthal/J. Muus
Alles über den Plus/4
1987, 373 Seiten
Das Buch enthält übersichtlich gegliedert alle Informationen, die für die praktische Arbeit am Computer notwendig sind.
Bestell-Nr. 90410, ISBN 3-89090-410-6
DM 39,- sFr 35,90/öS 304,20

W. Besenthal/J. Muus
Alles über den C16
1986, 292 Seiten
Bestell-Nr. 90385, ISBN 3-89090-385-1
DM 39,- sFr 35,90/öS 304,20

Ch. Spitzner
ROM-Listing C16, C116, Plus/4
1987, 436 Seiten
Bestell-Nr. 90425, ISBN 3-89090-425-4
DM 49,- sFr 45,10/öS 382,20

Commodore Sachbuchreihe
Alles über den C64
2. Auflage 1986, 514 Seiten
Dieses umfangreiche Grundlagenbuch enthält neben einem umfassenden BASIC-Lexikon alle Informationen und Tips, die der Spezialist zur Grafik- und Musikprogrammierung benötigt. Mit Anhang zum Betriebssystem GEOS.
Best.-Nr. 90379, ISBN 3-89090-379-7
DM 59,- sFr 54,30/öS 460,20

F. Riemenschneider
C64/C128 - Alles über Maschinensprache
1988, 320 Seiten, inkl. Diskette
Ein umfassendes Komplettwerk für die Entwicklung von eigenen Assemblerprogrammen auf dem Commodore 64. Alle benötigten Software-Voraussetzungen werden auf der beiliegenden Diskette mitgeliefert.
Bestell-Nr. 90571, ISBN 3-89090-571-4
DM 59,- sFr 54,30/öS 460,20

F. Müller
C64, Tips, Tricks und Tools
1988, 439 Seiten, inkl. Diskette
Tastatur-Tricks, Einzeiler, Peeks und Pokes, Basic-Routinen und Maschinenroutinen als Tuning für Ihre Programme, Toolkit-Programme mit allen Funktionen wie Merge, Renumber, Delete, Find, Auto und Dump. Programmoptimierung von Windows und komfortable Auswahlmenüs, Maskengenerator im Praxiseinsatz, effektives Programmieren in Assembler, Hilfs-, Monitor- und Kopierprogramme, Spiele-Pokes, Erstellen von Sicherheitskopien geschützer Programme.
Bestell-Nr. 90499, ISBN 3-89090-499-8
DM 59,- sFr 54,30/öS 460,20

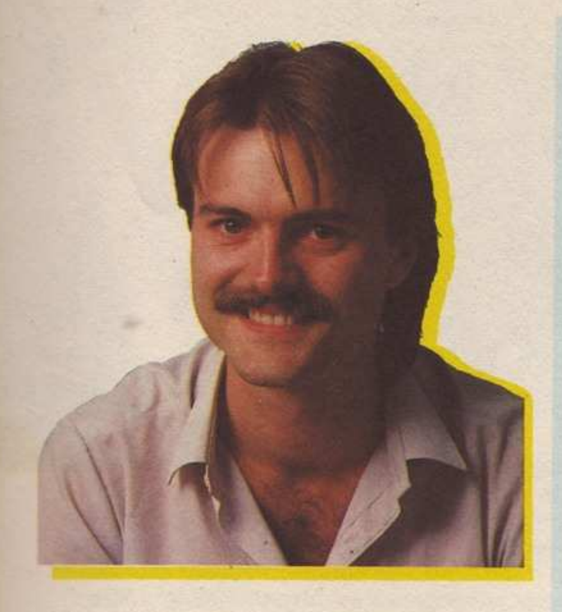


Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computerfachgeschäften oder in den Fachabteilungen der Warenhäuser.

133/806

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0

Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656 · ÖSTERREICH: Markt&Technik Verlag Gesellschaft m.b.H., Große Neugasse 28, A-1040 Wien, Telefon (0222) 5871393-0 · Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 · Ueberreuter Media Verlagsges. mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0.



KRISEN MANAGEMENT

Es gibt ein Schlagwort in der Computerbranche: Die Software-Krise. Zu Deutsch: Es gibt nie genau die Software, die man im Moment gerade braucht, und wenn doch, fehlt ihr genau die eine Funktion, die man so nötig bräuchte.

Für den C128 gilt dies in verstärktem Maße: Obwohl dieser Computer sehr leistungsfähig ist, wurde er von der Software-Industrie nie richtig angenommen - von einigen wenigen Textverarbeitungsprogrammen etc. abgesehen. Eine steile Qualitätsentwicklung der Software, die dann »alles aus der Kiste 'rausholt«, hat es von Seiten der Software-Industrie - mit Ausnahme von Geos 128 - nicht gegeben.

Von Seiten der C128-Fans aber sehr wohl: Sie entdecken immer neue Raffinessen in der Programmierung dieses Computers und die Programme die dabei entstehen, können sich im wahrsten Sinne des Wortes sehen lassen.

Mit unseren Sonderheften wollen wir den C128-Fans ein Podium bieten, eine Software-Börse, die allen C128-Besitzern diese Programme zugänglich macht - und die Software-Krise beseitigt.

Auch in dieser Ausgabe bieten wir Ihnen daher wieder einige äußerst interessante Programme an, die es für diesen Computer bislang nicht gab:

Das Haushaltsbuch erlaubt Ihnen, alle privaten Einnahmen und Ausgaben komfortabel zu verwalten. Übersichtliche Druckfunktionen lassen Sie dabei über den Stand Ihres Budgets nie im unklaren.

Wahrhaft professionell ist auch die Schachverwaltung, ein Programm, das schon auf verschiedenen großen Schachturnieren - so zum Beispiel bei den 15. Dortmunder Schachtagen - erfolgreich zur Turnierverwaltung und Auswertung herangezogen wurde.

Was geschickte Programmierer dem C128 entlocken können, zeigt Super-Vectors, eine Weiterentwicklung des Spiels, das Sie im Sonderheft 29 finden konnten. Es nutzt jetzt den 64 KByte großen Video-Speicher des neuen C128D im Blechgehäuse voll aus und wartet daher mit einer Grafikauflösung von sage und schreibe 736 x 354 Punkten auf. Das erreicht nicht einmal ein professioneller PC mit Hercules-Grafikkarte.

Ein Leckerbissen für Besitzer des 24-Nadel-Druckers NEC P 2200 ist Zenec 2.0, ein Zeichensatzeditor, mit dem Sie die tollsten Schriften für Ihren Drucker in Letter-Qualität entwerfen können. Auch dieses Programm besticht durch professionelles Software-Design.

Grundlagen zur Grafik- und Assembler-Programmierung, sowie über 25 Seiten der besten Tips & Tricks aus dem 64'er-Magazin runden dieses Sonderheft ab.

Klaus Schrödl
(Leitender Redakteur)

INHALT

Anwendungen

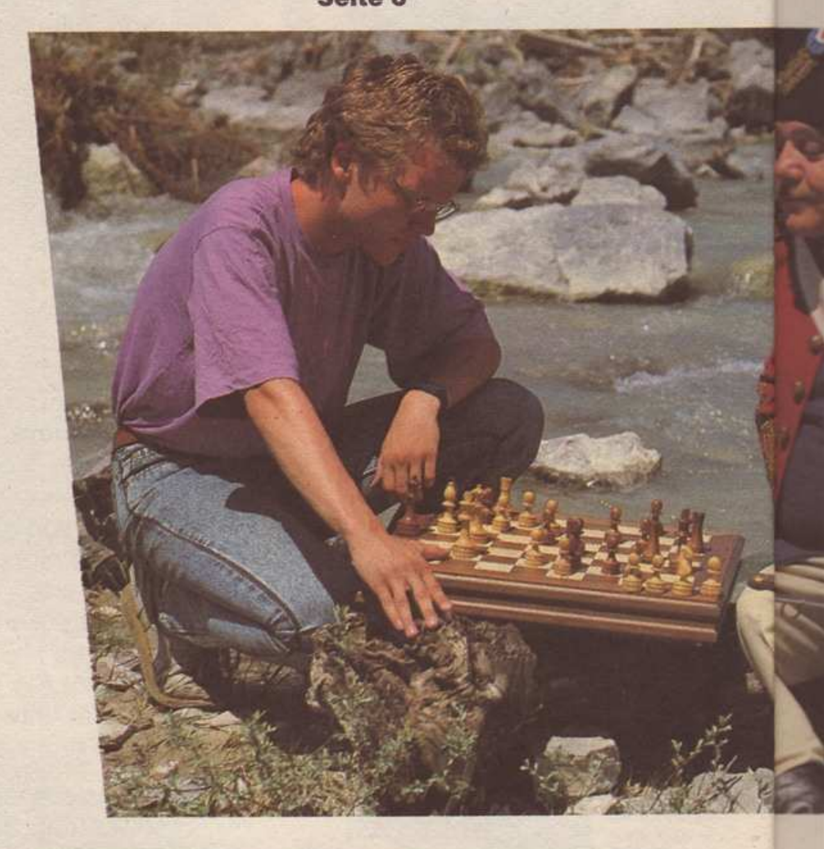
Grundlagen

Grafik

Tools



Ein paar Hunderter können bei geschickter Haushaltsführung schon überbleiben. Diese können dann für die »kleinen Annehmlichkeiten des Lebens« ausgeben. Das »Haushaltsbuch«, ein Top-Programm für den C128, hilft Ihnen dabei.
Seite 6



Drucker

Tips & Tricks

Foto: Space Press

Was bei dem C64 nur mit wüsten POKE-Orgien zu bewerkstelligen war, geht beim Basic 7.0 des C128 komfortabel und bequem: Die Programmierung von bewegten Grafikobjekten, den Sprites.
Seite 54

»Super-Vectors« – ein atemberaubendes und fesselndes Spiel: Ob Sie zu zweit spielen oder gegen den Computer antreten: Spannung ist garantiert. Speziell für den C128D mit 64 KByte Video-Speicher geschrieben, ist Super-Vectors das erste Spiel, das die enorme Bildschirm-Auflösung von 736 x 354 Punkten nutzt.
Seite 150

Schach – das Spiel der Könige – erfreut sich steigender Beliebtheit. In Schachturnieren messen sich die Anhänger dieses Sports in ihrem Können. Unser Programm »Chess-Tournament« ist ideal für die Verwaltung solcher Schachturniere. Die Ansetzung der Partien erfolgt nach dem Schweizer System.
Seite 18

Bücher

Spiele

Eingabehilfen

Sonstiges

Alle Programme aus Artikeln mit einem ■-Symbol finden Sie auch auf der Programmservice-Diskette zu diesem Sonderheft



Ihr Geld im Griff

Manchmal ist es gar nicht leicht, den Überblick über alle Einnahmen und Ausgaben zu behalten. Besitzer eines C 128 können jetzt aufatmen: Der Computer gibt mit Hilfe dieses Programms »Haushalt« eine detaillierte Übersicht über alle Buchungen aus, die innerhalb eines bestimmten Zeitraums anfallen.



ANWENDUNG

Bestimmt kennen Sie das auch: Da wird viel Geld für das Hobby ausgegeben, ein gelegentlicher Ausflug am Wochenende sollte zur Erholung auch sein. Jede Woche der Großeinkauf mit der Familie, und zur Entspannung gönnt man sich auch schon mal ein paar Bücher und Schallplatten. Und dann fragt man sich gegen Monatsende plötzlich, wo das ganze Geld geblieben ist ...

Mit dem Haushaltsbuch wird es nun anders. Alle anfallenden Buchungen können sehr einfach eingegeben werden. Eine Gesamtübersicht über die vorgenommenen Buchungen ermöglicht es, auf einen Blick zu erkennen, ob Sie bereits ins Soll gefallen sind oder noch über ein Guthaben verfügen.

Wer es schwarz auf weiß sehen will: Das Programm (Listing 1) besticht durch seine umfangreichen Druckausgaben.

Sollten Sie nach der Eingabe Fehlbuchungen feststellen, können Sie diese sehr schnell korrigieren. Einzelne Buchungen lassen sich über eingebaute Suchroutinen schnell nach verschiedenen Kriterien wie Datum oder Betrag auffinden.

Bis zu 500 Datensätze können im Programm gespeichert werden. Und selbstverständlich müssen feste und immer wiederkehrende Buchungen nicht immer wieder neu eingegeben werden: Einmal aufgenommen, werden sie immer wieder bei der Kalkulation berücksichtigt.

Neben diesen Funktionen bietet das Programm aber noch eine Vielzahl von kleinen Hilfen, die jeder Anwender zu schätzen wissen wird: ein kleines Diskettenmenü mit allen wichtigen Funktionen, einen eingebauten Rechner, der die vier Grundrechenarten sowie die Prozentrechnung beherrscht sowie einen Notizzettel mit einigen zusätzlichen Möglichkeiten.

Geben Sie das Listing 1 mit dem Checksummer 128 auf Seite 158 ein und speichern Sie es auf Diskette. Mit

RUN"HAUSHALT"

Markt&Technik

64'er SOFTWA[RE] EXT[RA]









64'er Extra: Grafik Vol. 1
Giga-CAD. Unschlagbare 3-D-Konstruktion auf dem C64. Hi-Eddi. Das Super-Zeichen- und -Malprogramm. Title Wizard. Giga-CAD-Filme für eigene Vorspänne. Pic-Loader. Verwenden Sie Hi-Eddi-Grafiken für eigene Programme. Hi-Maus. Maus-Treiber für Hi-Eddi. Hi-Spiegel. Spiegeln Sie beliebige Ausschnitte einer Grafik. Filmconverter. Giga-CAD-Filme können mit diesem Programm in das Hi-Eddi-Format umgewandelt werden. Druckeranpassungen für Hi-Eddi: Printer/Plotter VC 1520, MPS-801/802/803, Seikosha GP 700VC, Star NL-10, C.Itoh-8510, C.Itoh Riteman C+.

Bestell-Nr. 38701
DM 49,90* (sFr 44,90*/öS 499,-*)

64'er Extra: Grafik Vol. 2
Bestell-Nr. 38702
DM 39,90* (sFr 34,90*/öS 399,-*)

64'er Extra: Grafik Vol. 3
Bestell-Nr. 38703
DM 39,90* (sFr 34,90*/öS 399,-*)

Adventure-Pack Vol. 1
Robox: Fesselndes Grafik-Science-Fiction-Adventure. Der Herrscher eines fremden Planeten ließ sein Gehirn nach seinem Tod künstlich weiterleben – in einem Körper ohne Seele. Ihre Aufgabe ist es, zu Robox zu gelangen und ihn unschädlich zu machen, um die Erde vor dem sicheren Tod zu bewahren. Wie Sie es tun, bleibt Ihnen überlassen. Mit dem mitgelieferten Fall-Editor konstruieren Sie weitere Verbrechen und geben damit ihren Freunden harte Nüsse zu knacken. **Scotland Yard:** Spannendes Kriminal-Adventure. Begeben Sie sich auf spannende Verbrecherjagd in das London des 19. Jahrhunderts und lassen Sie sich bei Scotland Yard engagieren.
Mit dem mitgelieferten Fall-Editor konstruieren Sie weitere Verbrechen und geben damit Ihren Freunden harte Nüsse zu knacken.

Bestell-Nr. 38704
DM 29,90* (sFr 24,90*/öS 299,-*)

64'er Extra: Disk-Utilities Vol. 2
Disk-Mon 64: Professioneller Floppy- und Diskettenmonitor. Master-Copy: Backup-Kopierprogramm. Dual-Filecopy: File-Kopierprogramm für zwei Laufwerke. Track-Copy: Einfaches Kopieren und Formatieren von einzelnen Tracks. Tornado-Copy 1571: Schnelles Backup-Programm für einseitig bespielte Disketten. Hypra-Load/Save: Software-Speeder für C64. Hypra-Perfekt: Hypra-LOAD/SAVE, eingebunden ins Betriebssystem. EXOX V3: Leistungsfähiges Betriebssystem für C64. ProDisk: Komfortable Diskettenverwaltung in Assembler. EX.DIR & BAM: Ausführliches Directory. Hypra-Format 1541: Formatieren einer Diskette in nur 8 Sekunden. Disk-Searcher: Findet sehr schnell beliebige Zeichenketten auf Diskette. File-Manager: Befehlserweiterung zur Verwaltung von Disketten.

Bestell-Nr. 38707
DM 49,–* (sFr 44,-*/öS 490,-*)

64'er Extra: Disk-Utilities Vol. 1
Bestell-Nr. 38706
DM 49,–* (sFr 44,-*/öS 490,-*)

64'er Extra: Programmier-Utilities Vol. 1
Diese Sammlung leistungsfähiger Basic-Befehlserweiterungen ermöglicht es Ihnen, mit geringem Aufwand hochwertige Programme zu schreiben. **Hypra-Basic:** Mit dieser modular aufgebauten Befehlserweiterung wird es Ihnen möglich, je nach Anwendungsgebiet Befehle und Funktionen zusammenstellen. **Special-Basic:** Über 200 neue Basic-Befehle, die unter anderem die Bereiche Programmeditor, strukturierte Programmierung, komfortabler Zeichensatz, Sound sowie Ein-/Ausgabe- und Diskettenzugriffe umfassen, helfen Ihnen in fast allen Situationen, schnell und effektiv zu programmieren.

Bestell-Nr. 38716
DM 39,–* (sFr 35,-*/öS 390,-*)


Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0.
Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656,
ÖSTERREICH: Markt&Technik Verlag Gesellschaft m.b.H., Große Neugasse 28, A-1040 Wien, Telefon (0222) 5871393-0,
Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526,
Ueberreuter Media Verlagsges.mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (222) 481543-0

232/806



SOFTWARE DER EXTRAKLASSE









64'er Extra: C16 – C116 – Plus/4 MasterBase

Das Programm MasterBase bietet unter anderem folgende Möglichkeiten: Benutzerführung durch Pull-down-Menüs und Windows, maskierte Suchmöglichkeiten, indexsequentieller Zugriff, Reorganisation von zerstörten Datenbeständen, komfortabler Editor, u. a. zur Erstellung von Datei-Masken, Feldattributen etc., optimale Druckeranpassung, feldspezifische ESC-Sequenzen, Parameterdateien für seriellen und parallelen Druckerbetrieb, vorgefertigte und erweiterbare Code-Tabellen, Tastatur-Makros, Daten-Im- und -Export, Code-Wandlung von externen Dateien, Erstellung von Serienbriefen oder Rundschreiben.

Bestell-Nr. 38719

DM 49,–* (sFr 44,–*/öS 490,–*)

64'er Extra: Abenteuer-Spiele

Sein letzter Trick. Chicago zur Zeit der Prohibition: Beim Kartenspielen gewinnen Sie eine kleine Brennerei. Kurz vor der Ausführung eines großen Auftrags fliegt Ihre Brennerei auf. Bei einem Boß aus der Mitte Chicagos, Don Spazzatura, erstehen Sie »Ersatzmaterial«. Leider werden Sie von Spazzatura betrogen, er hat Ihnen nur Wasser verkauft! Jetzt schwören Sie sich nur eines: Rache für Don Spazzatura.

Wanderung. Irgendwann in ferner Zukunft: Sie sind der einzige Überlebende eines Raumschiffabsturzes. Ihre einzige Überlebenschance besteht darin, den nächsten Raumhafen zu erreichen – aber Sie müssen sich beeilen, denn Ihr Sauerstoffvorrat ist begrenzt...

Bestell-Nr. 38715

DM 39,–* (sFr 35,–*/öS 390,–*)

128'er Extra: The Best of 128'er

Hier finden Sie die besten Programme für den C128, die im 64'er Magazin und in den Sonderheften veröffentlicht werden. MASTERTEXT: Professionelle Textverarbeitung. COLOR-PACK 1: Super-Grafikerweiterung (480·240 Punkte Auflösung). TOP-FLOP: Leistungsfähiger Diskettenmonitor. DOUBLE-ASS: Zwei-Paß-Assembler, Unterstützung des Z80. WINDOW-TECH: Betriebssystem-Erweiterung, Unterstützung von 10 Windows. CENTRONICS-SCHNITTSTELLE: Unterstützung beliebiger CENTRONICS-Drucker. MICRO-HARDCOPY: Gestochen scharfe Hardcopys für Epson-Drucker und Kompatible. VECTORS: Super-Spiel im 80-Zeichen-Modus. UNIBOOT: Bootsektor manipulieren.

Bestell-Nr. 38712

DM 49,–* (sFr 44,–*/öS 490,–*)

128'er Extra: Paint R.O.I.A.L.

Paint R.O.I.A.L ist eines der wenigen Malprogramme, die die höchste Auflösung Ihres C128 verwenden. Wahlweise können sogar alle 16 Farben verwendet werden. Leistungsmerkmale: Auflösung: 640·200 Punkte (schwarz-weiß) 640·176 Punkte (Farbe), vielfältige Blockoperationen: Kopieren, Löschen, Laden, Speichern, Spiegeln, Rotieren, beliebiges Vergrößern und Verkleinern, wahlweise Ausführung aller Zeichenfunktionen mit Pinsel oder Sprühdose, definieren von Grafikfenstern, leistungsfähige Pinselfunktion mit frei definierbaren Pinselformen, Sprühdosenfunktion, kombinierbar mit den zwölf Pinselformen und Mustern, Radiergummi, UnDo-Funktion, Übernahme von C64-Bildern, Laden aus dem Directory.

Bestell-Nr. 38736

DM 49,–* (sFr 44,–*/öS 490,–*)

COMPUTER-LITERATUR UND SOFTWARE
HERBST/WINTER '88/'89

*Unverbindliche Preisempfehlung

Markt & Technik-Produkte erhalten Sie in den Fachabteilungen der Warenhäuser, im Versandhandel, in Computer-Fachgeschäften oder bei Ihrem Buchhändler

Fragen Sie Ihren Fachhändler nach unserem kostenlosen Gesamtverzeichnis mit über 500 aktuellen Computerbüchern und Software. Oder fordern Sie es direkt beim Verlag an!

können Sie das Programm von Diskette laden und starten. Danach meldet sich die Menü-Auswahl (Bild 1), von der aus drei weitere Untermenüs angewählt werden können.

Dieses Untermenü umfaßt neun Unterpunkte, die entweder durch die Eingabe des entsprechenden Buchstabens

Untermenü Haushaltsdatei

oder mit der Cursortaste angewählt und mit <SPACE> ausgeführt werden.

<a> Laden
Damit wird (nach einer Sicherheitsabfrage) eine Datei (Monat/Jahr) von Diskette geladen.

<b> Speichern
Eine vorher eingegebene Datei wird unter einem frei definierbaren Filenamen auf Diskette gespeichert. Beantwortet man die Sicherheitsabfrage mit <J>, wird ein bestehender Filename überschrieben.

<c> Übersicht
Hier werden alle laufenden Buchungen, Soll beziehungsweise Guthaben sowie die Gesamteinnahmen und Gesamtausgaben am Bildschirm angezeigt.



Bild 1. Das Hauptmenü mit seinen Unterfunktionen

<d> Eingabe
An dieser Stelle erfolgen die Eingaben für Abbuchungsart, den Buchungsbegriff, Betrag, Datum und Verrechnung (Bild 2). Das Feld Abbuchungsart hat dabei besondere Bedeutungen:
<s> steht für Sonstige Buchungen
<m> steht für Monatliche Buchungen
<d> steht für Doppelmonatliche Buchungen
<v> steht für Vierteljährliche Buchungen
<h> steht für Halbjährliche Buchungen
<j> steht für Jährliche Buchungen

Für das Eingabefeld <Begriff> sind maximal 18 Zeichen erlaubt. Bei gleichen Begriffen sollte die laufende Nummer mit angegeben werden, um Schwierigkeiten bei den Suchroutinen zu vermeiden.

Der Betrag kann ohne Nullen nach dem Komma eingegeben werden. Erlaubt sind maximal vierstellige Beträge.

Das Feld <Datum> bietet zwei Möglichkeiten der Eingabe: Entweder mit Punkteingabe (z.B. 01.01.88) oder ohne Punkteingabe (z.B. 010188). Beim zweiten Beispiel werden die Punkte anschließend automatisch eingefügt.

Das Feld <Art> bestimmt durch Eingabe von <+> oder <—> die Verrechnung als Einnahme oder Ausgabe.

Das Eingabemenü kann durch Eingabe eines Sternchens verlassen werden.

<e> Ausdrucken
Der Druckvorgang kann durch Drücken von <RUN/STOP> ohne Datenverlust unterbrochen werden. Allerdings muß dann das Blatt wieder neu justiert werden.

<f> Gesamtsuche
Als Suchkriterien sind alle Felder des Eingabemenüs zulässig. Dabei reichen bei eindeutigen Unterschieden die ersten zwei oder drei Zeichen aus. Ein Beispiel: Wir geben als Suchkriterium die Zeichen »Ba« ein. Auf dem Bildschirm werden dann alle Buchungen mit den Anfangszeichen »Ba« aufgelistet, also zum Beispiel »Bar 1«, »Bar 2«, »Banküberweisung«.

Von hier aus können Sie weitersuchen, den Notizzettel aufrufen oder ins Hauptmenü zurückkehren.

```
Haushaltsprogramm (C) 1987 by  W. Weber
Datum: TT.MM.JJJJ  Uhrzeit/Time: 000015
Eingabebegriff      ggf. mit lfn.Nr. 1
Freie Zeichen:     34806
Abbuchungsart      [ m ]
Begriff            [ Miete Wohnung       ]
Betrag             [   680.00 ]
Datum              [ 30.09.88 ]
Art +/- ?          [-]
Buchung korrekt [j/n]
30.09.88 m Miete Wohnung            680.00-
```

Bild 2. Nach jeder vollständigen Eingabe erfolgt eine Sicherheitsabfrage

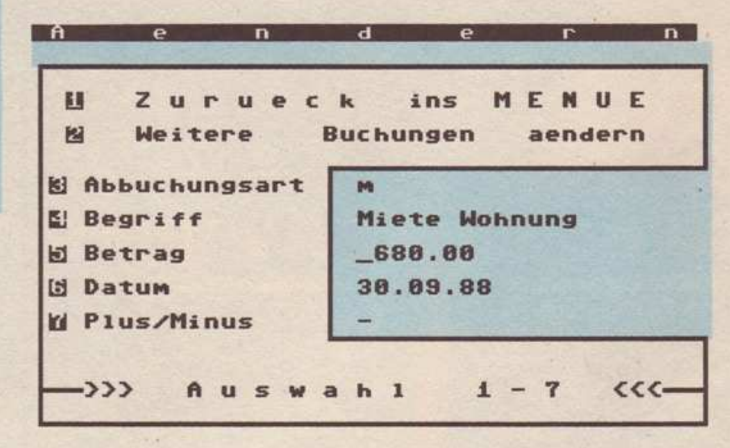


Bild 3. Korrekturen können auch nachträglich sehr einfach vorgenommen werden

<g> Ändern
Nach der Eingabe des Suchbegriffes erscheint eine Auswahl von sieben Menüpunkten (Bild 3). Jede einzelne Eingabe zum Suchbegriff kann gezielt geändert werden.

Untermenü Diskettenfunktionen

<f> Löschen
Sie geben einen Suchbegriff ein, und es wird nach einer Sicherheitsabfrage der entsprechende Datensatz gelöscht.

<i> Sortieren
Die Buchungen werden innerhalb eines Jahres nach Datum sortiert.

Die einzelnen Punkte dieses Menüs erklären sich von alleine. Sie können häufig benötigte Diskettenoperationen wie Anzeige des Directory, Dateien löschen oder umbenennen, Disketten formatieren oder validieren mit einem Tastendruck auslösen.

Dieses Untermenü bietet eine Reihe nützlicher Hilfen, die der Anwender zu schätzen wissen wird. Dazu gehört eine Anpassung der Bildschirmfarben an den individuellen Geschmack, eine Anzeige von Datum und Uhrzeit im

Untermenü Datum / Notizzettel

Hauptmenü, ein kleiner Rechner und ein Notizzettel mit einigen wichtigen Funktionen.

<o> Datum / Uhrzeit

Das Datum muß an dieser Stelle mit Punkten und der vierstelligen Jahreszahl eingegeben werden. Beispiel: 15.10.1988

Die Eingabe der Uhrzeit erfolgt ohne Punkte: 131705 bedeutet 13 Uhr, 17 Minuten und 5 Sekunden.

<p> Farbwahl

Mit den Zahlentasten <0> für Bildschirmhintergrund, <1> für Zeichenfarbe und <4> für Rahmenfarbe können



Bild 4. Neben dem aktivierten Taschenrechner kann jederzeit ein Hilfsschirm eingeblendet werden

Sie den Bildschirm nach Ihren eigenen Vorstellungen farblich anpassen. Drücken Sie die jeweilige Taste, bis die gewünschte Farbe erscheint. Bei Taste <1> erscheint dabei am unteren Bildschirmrand ein Farbbalken in der jeweiligen Zeichenfarbe.

<q> Notizzettel

Der Notizzettel bietet sechs Menüpunkte. Bevor Sie jedoch mit dem ersten Menüpunkt (Abbuchung auswählen) arbeiten können, sollten Sie im Programm (Listing 1) in den Datazeilen 5670 bis 6380 die leeren Begriffe mit festen Abbuchungen (Miete, Strom etc.) ergänzen. Für den Begriff dürfen Sie dabei maximal neun Zeichen verwenden. Beispiel:

```
5670 DATA"+ Gehalt... 2000   MTL"
5680 DATA"- Miete....  500   MTL"
```

Sollten die vorgegebenen Abbuchungen nicht ausreichen, müssen zusätzlich in Zeile 2 die Dimensionierungen der Variablen E1, L1$ und L2$ erhöht werden. Entsprechend kann hier verringert werden, wenn nicht soviele Abbuchungen benötigt werden. Eine Änderung der Feldgrößen erfordert schließlich noch eine Anpassung der FOR...TO...Anweisungen in den Zeilen 3630 und 5630.

Sind die Datazeilen geändert, wird beim Aufruf von <Abbuchung auswählen> der erste Datensatz eingelesen und auf dem Bildschirm angezeigt. Sie haben nun folgende Möglichkeiten:

<s>: Der angezeigte Datensatz wird gespeichert. Mit dieser Funktion gespeicherte Datensätze werden in den Notizzettel-Menüpunkten 2 und 3 aufgelistet.

<u>: Mit dieser Funktion kann der angezeigte Datensatz übersprungen werden.

<n>: Sollte eine Abbuchung, die nur sehr selten vorkommt, für eine Kalkulation benötigt werden, kann sie mit diesem Menüpunkt zusätzlich eingefügt werden. Diese Einfügungen gehen bei Programmende oder bei erneutem Aufruf von <Abbuchungen auswählen> wieder verloren.

<m>: Mit dieser Taste springen Sie ins Notizzettel-Menü zurück.

Wählen Sie von dort aus den zweiten Menüpunkt <Abbuchungen auflisten>, werden alle Datensätze aufgelistet, die Sie vorher mit <s> oder <n> ausgewählt haben. Die einzelnen Beträge werden subtrahiert oder addiert und im Ergebnis als Soll oder Guthaben angezeigt. Sind mehr Einträge vorhanden, können Sie zurückblättern, die Liste ausdrucken, den Taschenrechner aktivieren oder zum Notizzettel-Menü zurückkehren.

Unter dem Menüpunkt <Gesamtkalkulation> haben Sie die Möglichkeit, alle Werte in übersichtlicher Form auf dem Bildschirm oder auf dem Drucker dargestellt zu sehen. Nach einer Auflistung aller ausgesuchten Abbuchungen erscheint eine Aufstellung des Kalkulations-Guthabens und des aktuellen Solls, aus dem ein Gesamtguthaben oder Gesamtsoll errechnet wird.

Der Aufruf des Taschenrechners (Bild 4) erlaubt die Verwendung der vier Grundrechenarten sowie der Prozentrechnung. Eine Übersicht über die wichtigsten Tasten finden Sie in Tabelle 1.

Achtung: Bei der Division durch null erfolgt eine Fehlermeldung. Nach diesem Abfangen des Fehlers können Sie ganz normal weiterarbeiten.

Der fünfte Menüpunkt des Notizzettels <Suchroutine> erlaubt eine rasche Kontrolle, ob eine eingelesene Buchung vielleicht schon erledigt ist.

Den Sprung zurück ins Hauptmenü bewirkt der letzte Menüpunkt.

<r> Ende / Hilfe

Die Wahl dieses Menüpunktes gibt entweder einige kleine Hilfen auf dem Bildschirm aus oder beendet das Programm. Nach einem Verlassen des Programms kann mit

```
GOTO 5
```

ohne Datenverlust wieder gestartet werden.



<c>	= Letzte Eingabe löschen
<a>	= Neustart
<s>	= Ergebnisse zwischenspeichern
<m>	= mit <s> gesicherte Ergebnisse abrufen
<l>	= Löscht den gesamten Rechnerinhalt
<z>	= Zurück ins Hauptmenü
<p>	= Abbuchungen auflisten, Soll/Guthaben errechnen
<%>	= Prozentrechnen
<o>	= Rücksprung ins Notizzettel-Menü
<h>	= Ergebnis Soll/Guthaben (aktuelle Datei) abrufen
<i>	= Aktiviert <info> als Hilfe
<o>	= Schaltet Rechner aus

Tabelle 1. Die wichtigsten Funktionen des Taschenrechners im Überblick

Hinweise zum Programm

Das Programm ist vollständig in Basic geschrieben und nutzt je nach Voreinstellung den 40- oder 80-Zeichen-Bildschirm. Der Geschwindigkeitsvorteil ist im Fast-Modus bei 80-Zeichen-Darstellung enorm.

(Wilfried Weber/ef)

```
10 PRINT "{2HOME}": IF RWINDOW(2)=80 THEN FAST :
   ELSE SLOW                                              <J2Q>
20 PRINT CHR$(14): COLOR 5,16: COLOR 0,1: COLOR 4
   ,1: SCNCLR : DA$="TT.MM.JJJJ": POKE 808,112: P
   OKE 792,98                                             <9D8>
30 DIM N$(500),S$(500),P$(500),T$(500),U$(500),R$
   (20),E1(73),L1$(72),L2$(72): TRAP 6910                 <TIP>
40 LP$="-{16SPACE}-": GOTO 60                             <62C>
50 PRINT "{2DOWN,6RIGHT}ZURUECK INS MENUE MIT TAS
   TE": GET KEY A$: SCNCLR                                <NLU>
60 SCNCLR : OPEN 1,4,7: CLOSE 1: W=1: PUDEF " ":
   D$=" DM": MP$="{10SPACE}"                              <KHK>
70 L8$="-{38SPACE}-                                       <VA9>
80 L7$="7**************************************X"         <PP1>
90 L6$="R**************************************S"         <CLI>
100 L4$="****************************************
    "                                                     <BI6>
110 L9$="S**************************************W
    "                                                     <G5L>
120 L5$=" +++++++++++++++++++++++++++++++++++++++
    ++++++++++++++++++++++++++++++"                       <8EH>
130 L$="{15SPACE}"                                        <G5Q>
140 PRINT "{2HOME,CLR,RVSON} HAUSHALTSVERWALTUNG
    1987 (C){3SPACE}H.WEBER {RVOFF}"                      <RLR>
150 PRINT "{RVSON} DATUM: ";DA$;TAB(18)"{6SPACE}U
    HRZEIT: ";TI$;TAB(39)"{RVSON} {RVOFF}": POKE
    229,2                                                 <4T1>
160 PRINT "{HOME,17SPACE}R**********************S":
     R$(W)="{RVSON}"                                      <KF2>
170 PRINT " R****************ESDISKETTENFUNKTIONEN
    -"                                                    <C72>
180 PRINT " -R**************SS********************
    W"                                                    <7J4>
190 PRINT " --{RVSON,14SPACE,RVOFF}--"R$(10)"-J-
    FILES LOESCHEN {RVOFF}-"                              <B4T>
200 PRINT " --{RVSON,3SPACE}M E{SHF.SPACE}N{SHF.S
    PACE}U{SHF.SPACE}E{2SPACE,RVOFF}--"R$(11)"-K-
     FILES RENAME{3SPACE,RVOFF}-"                         <9II>
210 PRINT " --{RVSON,14SPACE,RVOFF}--"R$(12)"-L-
    VALIDIEREN{5SPACE,RVOFF}-"                            <JIQ>
220 PRINT " --{RVSON,4SPACE}AUSWAHL{3SPACE,RVOFF}
    --"R$(13)"-M- FORMATIEREN{4SPACE,RVOFF}-"             <P23>
230 PRINT " --{RVSON,14SPACE,RVOFF}--"R$(14)"-N-
    DIRECTORY{6SPACE,RVOFF}-"                             <KMQ>
240 PRINT " -7***************XS*******************
    W"                                                    <BQM>
250 PRINT " S****************RXDATUM / NOTIZZETTEL
    -"                                                    <M8J>
260 PRINT " -HAUSHALTSDATEI S*********************
    W"                                                    <GJR>
270 PRINT " S****************W"R$(15)"-O- DATUM /
    UHRZEIT {RVOFF}-"                                     <221>
280 PRINT " -" TAB(2)R$(1)"-A- LADEN{6SPACE,RVOFF
    }-"R$(16)"-P- FARBWAHL{8SPACE,RVOFF}-"                <1D1>
290 PRINT " -" TAB(2)R$(2)"-B- SPEICHERN{2SPACE,R
    VOFF}-"R$(17)"-Q- NOTIZZETTEL{5SPACE,RVOFF}-"         <HDG>
300 PRINT " -" TAB(2)R$(3)"-C- UEBERSICHT {RVOFF}
    -"R$(18)"-R- ENDE / HILFE{4SPACE,RVOFF}-"             <9PF>
310 PRINT " -" TAB(2)R$(4)"-D- EINGABE{4SPACE,RVO
    FF}S**********************W"                          <D0G>
320 PRINT " -" TAB(2)R$(5)"-E- AUSDRUCKEN {RVOFF}
    -{RVSON,20SPACE,RVOFF}-"                              <O7T>
330 PRINT " -" TAB(2)R$(6)"-F- GESAMTSUCHE{RVOFF}
    -{RVSON,2SPACE}CURSOR UND SPACE{2SPACE,RVOFF}
    -"                                                    <9H1>
340 PRINT " -" TAB(2)R$(7)"-G- AENDERN{4SPACE,RVO
    FF}-{RVSON,20SPACE,RVOFF}-"                           <N09>
350 PRINT " -" TAB(2)R$(8)"-H- LOESCHEN{3SPACE,RV
    OFF}-{RVSON,2SPACE}ODER{2SPACE}BUCHSTABEN{2SP
    ACE,RVOFF}-"                                          <H0C>
360 PRINT " -" TAB(2)R$(9)"-I- SORTIEREN{2SPACE,R
    VOFF}-{RVSON,20SPACE,RVOFF}-"                         <4KS>
370 PRINT " 7****************P*********************
    X"                                                    <3LO>
380 DO : PRINT "{2HOME}";: CHAR ,33,1,TI$,1               <1SN>
390 GET A$: LOOP WHILE A$="": R$(W)="": POKE 229,
    2                                                     <AOC>
400 A=ASC(A$)-64: IF A<1 OR A>18 THEN 420                 <4F3>
410 ON A GOTO 500,640,770,1020,1340,1620,1820,232
    0,2520,6390,6460,6530,6580,2910,6870,6680,339
    0,3150                                                <CT0>
420 IF A$=" " THEN 480                                    <ORM>
430 IF A$="{DOWN}" THEN W=W+1+9*(W=9 OR W=18)             <BSB>
440 IF A$="{UP}" THEN W=W-1-9*(W=1 OR W=10)               <1S1>
450 IF A$="{RIGHT}" THEN W=W+9+18*(W>9)                   <BNH>
460 IF A$="{LEFT}" THEN W=W-9-18*(W<10)                   <INM>
470 GOTO 160                                              <R20>
480 R$(W)=""                                              <3EL>
490 ON W GOTO 500,640,770,1020,1340,1620,1820,232
    0,2520,6390,6460,6530,6580,2910,6870,6680,339
    0,3150                                                <7L1>
500 PRINT "{2HOME,RVSON,CLR,6SPACE}L{6SPACE}A{6SP
    ACE}D{6SPACE}E{6SPACE}N{5SPACE,RVOFF}"                <3C7>
510 CHAR ,5,10,"DATEI LADEN{2SPACE}(J/N) ": GET K
    EY A$                                                 <DP4>
520 IF A$="J" THEN 540                                    <MBP>
530 IF A$="N" THEN 60: ELSE 510                           <JLU>
540 CHAR ,5,10,"{RVSON} N A M E{2SPACE}DER DATEI
    {RVOFF} ": INPUT F$                                   <BA0>
550 OPEN 1,8,0,F$+",S,R"                                  <HHL>
560 INPUT#1,N                                             <9SL>
570 FOR I=1 TO N                                          <IBS>
580 INPUT#1,N$(I),S$(I),P$(I),T$(I),U$(I)                 <J3F>
590 NEXT I                                                <JI4>
600 CHAR ,3,10,"{39SPACE}"                                <Q1C>
610 CHAR ,3,10,"ES WURDEN": PRINT N;: PRINT "BUCH
    UNGEN EINGELADEN ."                                   <3Q2>
620 CLOSE 1                                               <0CN>
```

```
630 GOTO 50                                                <F9B>
640 PRINT "(2HOME,RVSON,CLR,3SPACE)S(3SPACE)P(3SP
    ACE)E(3SPACE)I(3SPACE)C(3SPACE)H(3SPACE)E(3SP
    ACE)R(3SPACE)N(4SPACE,RVOFF)"                          <FDB>
650 CHAR ,5,10,"NAME DER DATEI ": INPUT F$                 <9PM>
660 CHAR ,5,10,"(RVSON) ALTE DATEI ZERSTOEREN (J/
    N)(RVOFF) ": GET KEY A$                                <A9I>
670 IF A$<>"J" THEN 60                                     <LQB>
680 OPEN 1,8,1,"@:"+F$+",S,W"                              <C19>
690 PRINT#1,N                                              <B45>
700 FOR I=1 TO N                                           <JRS>
710 PRINT#1,N$(I): PRINT#1,S$(I): PRINT#1,P$(I):
    PRINT#1,T$(I): PRINT#1,U$(I)                           <LK1>
720 NEXT I                                                 <BT4>
730 CHAR ,3,10,"(39SPACE)"                                 <014>
740 CHAR ,0,10,"ES WURDEN": PRINT N;: PRINT "BUCH
    UNGEN EINGESPEICHERT ."                                <V3D>
750 CLOSE 1                                                <G6N>
760 GOTO 50                                                <V7B>
770 REM                                                    <GA1>
780 PRINT "(2HOME,CLR,RVSON,2SPACE)GESAMTEINNAHME
    N(2SPACE)UND(2SPACE)GESAMTAUSGABEN(2SPACE,RVO
    FF)"                                                   <KRT>
790 POKE 229,3                                             <HNF>
800 SCNCLR : SS=0                                          <ORT>
810 FOR I=1 TO N: S=VAL(S$(I))                             <CT0>
820 PRINT P$(I);TAB(9);U$(I);TAB(11);N$(I);TAB(31
    );: PRINT USING "####.##";S;                           <OLT>
830 PRINT TAB(39);T$(I)                                    <FR5>
840 IF INT(I/8)<>I/8 THEN 900                              <BG7>
850 PRINT "(2DOWN)WAHL W-(RVSON)WEITER(RVOFF) *-(
    RVSON)MENUE(RVOFF) A-(RVSON)ANFANG(RVOFF) Z=(
    RVSON)ZURUECK(RVOFF)": PRINT : PRINT                   <QTV>
860 GET KEY A$: IF A$="W" THEN 890                         <BMD>
870 IF A$="A" THEN 800                                     <JB8>
880 IF A$="*" THEN 60: ELSE IF A$="Z" THEN SCNCLR
     : I=I-16: ELSE 860                                    <JGV>
890 SCNCLR                                                 <BDJ>
900 IF I>0 THEN NEXT I: ELSE GOTO 800                      <BFB>
910 IF SS>0 AND(SO<>0 OR EI<>0 OR SU>0) THEN 920           <E67>
920 GOSUB 7010: PRINT L4$                                  <EML>
930 IF SU>0 THEN PRINT "(DOWN,13SPACE)GUTHABEN(8S
    PACE)";: PRINT USING "######.##";SU                    <EUM>
940 IF SU<0 THEN PRINT "(DOWN,17SPACE)SOLL(8SPACE
    )";: PRINT USING "######.##";SU                        <VCN>
950 PRINT L4$                                              <22E>
960 PRINT "(DOWN) GESAMTEINNAHMEN";: PRINT TAB(22
    );: PRINT USING "######.##";EI                         <NDI>
970 PRINT "(DOWN) GESAMTAUSGABEN ";: PRINT TAB(22
    );: PRINT USING "######.##";SO                         <BDA>
980 PRINT L4$                                              <1AE>
990 PRINT "(2DOWN)AUSWAHL(6SPACE)*-MENUE(4SPACE)A
    -ANFANG": PRINT : PRINT                                <2JJ>
1000 GET KEY A$: IF A$="*" THEN 60                         <OD0>
1010 IF A$="A" THEN 800: ELSE 1000                         <7GU>
1020 REM                                                   <3KE>
1030 N=N+1                                                 <M2B>
1040 PRINT "(2HOME,CLR,RVSON) HAUSHALTSPROGRAMM (
     C) 1987 BY(2SPACE)W. WEBER(RVOFF)"                    <5MJ>
1050 PRINT "(RVSON) DATUM: ";DA$;TAB(18)"(2SPACE)
     UHRZEIT/TIME: ";TI$                                   <HFM>
1060 PRINT L6$: PRINT "-EINGABEBEGRIFF(3SPACE)GGF
     . MIT LFN.NR.";N;TAB(39)"=": PRINT L8$                <023>
1070 PRINT "=FREIE ZEICHEN: ";FRE(0);TAB(39)"=":
     PRINT L9$                                             <BB3>
1080 FOR U=1 TO 16: PRINT L8$: NEXT U: PRINT L7$           <380>
1090 CHAR ,1,19,"**********************************
     ******"                                               <51R>
1100 CHAR ,1,20,"(RVSON)ABBUCHUNGS-: S-SONSTIGE(3
     SPACE)M-MONATLICH (RVOFF)"                            <P78>
1110 CHAR ,1,21,"(RVSON)ARTEN(8SPACE)D-DOPPELMON.
      V-1/4JAEHRL.(RVOFF)"                                 <JCR>
1120 CHAR ,1,22,"(RVSON) * = MENUE(3SPACE)H-1/2JA
     EHRL. J-JAEHRLICH (RVOFF)"                            <LQC>
1130 GOSUB 2920                                            <LNH>
1140 IF V$="*" THEN N=N-1: GOTO 60                         <6LO>
1150 U$(N)=LEFT$(Q$,1)                                     <OMG>
1160 GOSUB 2930                                            <NFG>
1170 N$(N)=LEFT$(Q$,18)                                    <KPN>
1180 GOSUB 2680                                            <L75>
1190 S$(N)=LEFT$(Q$,7)                                     <VUD>
1200 GOSUB 2780                                            <IN7>
1210 P$(N)=LEFT$(D$,8)                                     <HFN>
1220 GOSUB 2860                                            <NNN>
1230 PRINT "(2DOWN)R**********************************
     *********"                                            <503>
1240 PRINT "=(RVSON)BUCHUNG KORREKT [J/N](16SPACE
     ,RVOFF)"                                              <475>
1250 PRINT "=(RVSON,38SPACE,RVOFF)"                        <E6N>
1260 PRINT TAB(1);: PRINT USING "(RVSON)#########
     ";P$(N);                                              <0G3>
1270 PRINT TAB(10);: PRINT USING "##";U$(N);               <7KF>
1280 PRINT TAB(11);: PRINT USING "################
     ####";N$(N);                                          <MC9>
1290 S=VAL(S$(N)): PRINT TAB(30);: PRINT USING "#
     ###.##";S;: PRINT M$                                  <NUJ>
1300 GET KEY A$                                            <60D>
1310 IF A$="J" THEN T$(N)=LEFT$(M$,1): GOTO 1030           <GV0>
1320 LO$=CHR$(32)                                          <819>
1330 IF A$="N" THEN P$(N)=LO$: S$(N)=CHR$(48): N$
     (N)=LO$: U$(N)=LO$: GOTO 1040: ELSE 1300              <B9P>
1340 POKE 808,110: SCNCLR                                  <6J4>
1350 PRINT CHR$(14): OPEN 1,4,7: CMD 1: PRINT              <JOJ>
1360 PRINT " H A U S H A L T S P R O G R A M M(3S
     PACE)BY WILFRIED WEBER(2SPACE)MONAT ";F$              <VR0>
1370 PRINT " TAGESDATUM : ";DA$;TAB(23)"(2SPACE)U
     HRZEIT/TIME: ";TI$                                    <0GK>
1380 PRINT : PRINT : SU=0: S=0: EI=0: SO=0: F=0:
     SS=0                                                  <ADA>
1390 PRINT " BUCHUNGSARTEN : -S-SONSTIGE.....---M
     -MONATL...---D-DOPPELMTL BUCHUNG                      <VMR>
1400 PRINT "(17SPACE)-V-1/4JAEHRLICH ---H-1/2JAEH
     RL---J-JAEHRL.(3SPACE)BUCHUNG": PRINT                 <JB3>
1410 PRINT " (RVSON) DATUM(6SPACE)ART(6SPACE)BEGR
     IFF(26SPACE)BETRAG(4SPACE)+/- ": PRINT                <SIM>
1420 FOR I=1 TO N                                          <JNT>
1430 S=VAL(S$(I))                                          <D0M>
1440 PRINT TAB(1);: PRINT USING "########";P$(I);
     : PRINT TAB(5);: PRINT USING "#";U$(I);               <UVD>
1450 PRINT TAB(7);: PRINT USING "################
     ####";N$(I);                                          <N60>
1460 PRINT TAB(12);: PRINT USING "####.##";S;              <JJ6>
1470 PRINT TAB(5);: PRINT USING "####.##";T$(I)            <RJA>
1480 M$=MID$(T$(I),1,1)                                    <KB6>
1490 I$=STR$(S)                                            <67L>
1500 IF M$="-" THEN SU=SU-VAL(I$): ELSE SU=SU+VAL
     (I$)                                                  <404>
1510 IF M$="-" THEN SO=SO+VAL(I$): ELSE EI=EI+VAL
     (I$)                                                  <K05>
1520 NEXT I                                                <DT4>
1530 IF SS>0 AND(SO<>0 OR EI<>0 OR SU>0) THEN 154
     0                                                     <67S>
1540 PRINT L5$                                             <L6F>
1550 IF SU>0 THEN PRINT TAB(26)"GUTHABEN(2SPACE)"
     ;TAB(5);: PRINT USING "#####.##";SU                   <1CQ>
1560 IF SU<1 THEN PRINT TAB(29)"SOLL(3SPACE)";TAB
     (5);: PRINT USING "#####.##";SU                       <JRE>
1570 PRINT L5$                                             <LUF>
1580 PRINT "(11SPACE)GESAMTEINNAHMEN";: PRINT TAB
     (26);: PRINT USING "######.##";EI;: PRINT             <0TK>
1590 PRINT "(11SPACE)GESAMTAUSGABEN ";: PRINT TAB
     (26);: PRINT USING "######.##";SO: PRINT L5$          <H1H>
1600 I=N: F=-1                                             <5K7>
1610 GOTO 60                                               <3T8>
1620 PRINT "(2HOME,CLR,RVSON,5SPACE)S(5SPACE)U(5S
     PACE)C(5SPACE)H(5SPACE)E(5SPACE)N(4SPACE,RVO
     FF)": POKE 229,2: PRINT : PRINT                       <QE1>
1630 INPUT "(UP)> SUCHBEGRIFF < ";W$: PRINT                <CK0>
1640 F=0: SO=0: S=0: EI=0                                  <CPH>
1650 FOR I=1 TO N                                          <I7T>
1660 IF LEFT$(N$(I),LEN(W$))=W$ THEN 1710                  <4S5>
1670 IF LEFT$(S$(I),LEN(W$))=W$ THEN 1710                  <707>
1680 IF LEFT$(P$(I),LEN(W$))=W$ THEN 1710                  <04P>
1690 IF LEFT$(T$(I),LEN(W$))=W$ THEN 1710                  <04R>
1700 IF LEFT$(U$(I),LEN(W$))=W$ THEN 1710: ELSE 1
     750                                                   <SB8>
1710 S=VAL(S$(I))                                          <T0M>
1720 PRINT P$(I);TAB(9);U$(I);TAB(11);N$(I);TAB(3
     1);: PRINT USING "####.##";S;                         <EFT>
1730 PRINT TAB(39);: PRINT USING "####.##";T$(I):
      M$=MID$(T$(I),1,1): I$=STR$(S)                       <QNQ>
1740 IF M$="-" THEN SO=SO+VAL(I$): ELSE EI=EI+VAL
     (I$)                                                  <M05>
1750 NEXT I                                                <LU4>
1760 PRINT L4$: PRINT "(DOWN) GESAMTEINNAHMEN";:
     PRINT TAB(22);: PRINT USING "######.##";EI            <ERE>
1770 PRINT "(DOWN) GESAMTAUSGABEN ";: PRINT TAB(2
     2);: PRINT USING "######.##";SO: I=N: F=-1            <OC8>
1780 PRINT "(2DOWN,3SPACE)W=WEITER(3SPACE)*=MENUE
     (3SPACE)N=NOTIZZETTEL"                                <NM0>
1790 GET KEY A$: IF A$="W" THEN 1620                       <JLR>
1800 IF A$="*" THEN 60                                     <3M6>
1810 IF A$="N" THEN 3390                                   <R19>
1820 PRINT "(2HOME,CLR,RVSON) A(5SPACE)E(5SPACE)N
     (5SPACE)D(5SPACE)E(5SPACE)R(5SPACE)N (RVOFF)
     ": POKE 229,1                                         <CIK>
1830 INPUT "(2DOWN) > SUCHBEGRIFF < ";W$: F=0: SC
     NCLR                                                  <T3J>
1840 FOR I=1 TO N                                          <SNT>
1850 IF LEFT$(N$(I),LEN(W$))=W$ THEN 1900                  <C56>
1860 IF LEFT$(S$(I),LEN(W$))=W$ THEN 1900                  <E10>
1870 IF LEFT$(P$(I),LEN(W$))=W$ THEN 1900                  <ET2>
1880 IF LEFT$(T$(I),LEN(W$))=W$ THEN 1900                  <9TS>
1890 IF LEFT$(U$(I),LEN(W$))=W$ THEN 1900: ELSE 2
     290                                                   <7B2>
1900 CHAR ,0,1,L6$                                         <LIC>
1910 CHAR ,0,2,L8$                                         <K2Q>
1920 CHAR ,0,3,"= (RVSON)1(RVOFF,3SPACE)Z U R U E
      C K(3SPACE)INS(2SPACE)M(SHF.SPACE)E(SHF.SPA
     CE)N(SHF.SPACE)U(SHF.SPACE)E(3SPACE)="                <LM6>
1930 CHAR ,0,4,L8$                                         <G3A>
1940 CHAR ,0,5,"= (RVSON)2(RVOFF,3SPACE)WEITERE(4
     SPACE)BUCHUNGEN(3SPACE)AENDERN(3SPACE)="              <6T5>
1950 CHAR ,0,6,L8$                                         <C3Q>
1960 CHAR ,0,7,"=(16SPACE)R***********************V
     "                                                     <0FE>
1970 CHAR ,0,8,"=(RVSON)3(RVOFF) ABBUCHUNGSART ="
     : CHAR ,19,8,"(3SPACE)": CHAR ,19,8,U$(I)             <61G>
1980 CHAR ,0,9,LP$                                         <S7A>
1990 CHAR ,0,10,"=(RVSON)4(RVOFF) BEGRIFF(7SPACE)
     =": CHAR ,19,10,"(20SPACE)": CHAR ,19,10,N$(
     I)                                                    <QPK>
2000 CHAR ,0,11,LP$                                        <571>
2010 S=VAL(S$(I)): PUDEF "B": CHAR ,0,12,"=(RVSON
     )5(RVOFF) BETRAG(8SPACE)=": CHAR ,19,12,"(10
     SPACE)"                                               <4KK>
2020 PRINT USING "(10LEFT)####.##";S                       <EKB>
2030 CHAR ,0,13,LP$                                        <P7H>
2040 CHAR ,0,14,"=(RVSON)6(RVOFF) DATUM(9SPACE)="
     : CHAR ,19,14,"(10SPACE)": CHAR ,19,14,P$(I)          <I0U>
```

Listing 1. Mit »Haushalt« behalten Sie die Übersicht. Bitte geben Sie das Listing mit dem Checksummer auf Seite 158 ein.

```
2050 CHAR ,0,15,LP$                                    <521>
2060 CHAR ,0,16,"-(RVSON)7(RVOFF) PLUS/MINUS(4SPA
     CE)-": CHAR ,19,16,T$(I)                          <PDQ>
2070 CHAR ,0,17,"-(16SPACE)Z*********************
     S"                                                <HSA>
2080 CHAR ,0,18,L8$                                    <DE0>
2090 CHAR ,0,19,L8$                                    <7E8>
2100 CHAR ,0,20,"W**>>>(3SPACE)A U S W A H L(4SPA
     CE)1 - 7(3SPACE)<<<**W"                           <CNS>
2110 CHAR ,0,21,L8$                                    <B5P>
2120 CHAR ,0,22,L7$                                    <MTE>
2130 GET KEY A$: A=VAL(A$): IF A<1 OR A>7 THEN 21
     30                                                <I7D>
2140 ON A GOTO 60,2280,2150,2170,2190,2210,2230        <ADJ>
2150 GOSUB 2920                                        <GNH>
2160 U$(I)=LEFT$(Q$,1): GOTO 1970                      <6K5>
2170 GOSUB 2930                                        <HVG>
2180 N$(I)=LEFT$(Q$,18): GOTO 1990                     <SK6>
2190 GOSUB 2680                                        <JN5>
2200 S$(I)=LEFT$(Q$,7): GOTO 2010                      <24T>
2210 GOSUB 2780                                        <K77>
2220 P$(I)=LEFT$(D$,8): GOTO 2040                      <ICP>
2230 CHAR ,1,16,"ART +/- ?(7SPACE)[(3SPACE)]           <CBU>
2240 GET KEY A$                                        <4GD>
2250 IF A$="+" THEN PRINT "(3LEFT)+(2SPACE)": M$=
     "+": GOTO 2270                                    <EGD>
2260 IF A$="-" THEN PRINT "(3LEFT)-(2SPACE)": M$=
     "-": GOTO 2270: ELSE 2240                         <4G8>
2270 T$(I)=MID$(M$,1,1): GOTO 2060                     <B2A>
2280 I=N: F=-1                                         <6K7>
2290 NEXT I                                            <5K4>
2300 IF NOT F THEN PRINT " (2DOWN)BEGRIFF ";W$;".
      NICHT VORHANDEN !!!": GOTO 50                    <ME0>
2310 GOTO 1820                                         <7G6>
2320 PRINT "(2HOME,CLR,RVSON,3SPACE)L(3SPACE)O(4S
     PACE)E(4SPACE)S(4SPACE)C(4SPACE)H(4SPACE)E(4
     SPACE)N(2SPACE,RVOFF)": POKE 229,1                <FRO>
2330 INPUT "(2DOWN)> SUCHBEGRIFF < ";W$: F=0: FOR
      I=1 TO N                                         <GG0>
2340 IF LEFT$(N$(I),LEN(W$))=W$ THEN 2390              <5S1>
2350 IF LEFT$(S$(I),LEN(W$))=W$ THEN 2390              <603>
2360 IF LEFT$(P$(I),LEN(W$))=W$ THEN 2390              <745>
2370 IF LEFT$(T$(I),LEN(W$))=W$ THEN 2390              <747>
2380 IF LEFT$(U$(I),LEN(W$))=W$ THEN 2390: ELSE 2
     490                                               <NRE>
2390 SCNCLR : PRINT "(UP,37SPACE)": S=VAL(S$(I)):
      PRINT "(2UP)";L6$                                <0D7>
2400 PRINT "-";CHR$(18);P$(I);TAB(10);U$(I);TAB(1
     2);N$(I);TAB(31);: PRINT USING "####.##";S;       <4G8>
2410 PRINT TAB(38);T$(I);TAB(39)"(RVOFF)-": PRINT
      L9$: PRINT L8$: PRINT L8$: PRINT TAB(13)"(U
     P)LOESCHEN (J/N)": PRINT L7$                      <OJC>
2420 GET KEY A$: IF A$="N" THEN 60                     <UT0>
2430 IF A$="J" THEN 2440: ELSE 2420                    <8MG>
2440 FOR Z= I TO N-1                                   <7C1>
2450 N$(Z)=N$(Z+1): S$(Z)=S$(Z+1): P$(Z)=P$(Z+1):
      T$(Z)=T$(Z+1): U$(Z)=U$(Z+1)                     <R1D>
2460 NEXT Z                                            <VRD>
2470 N=N-1                                             <NID>
2480 I=N: F=-1                                         <5K7>
2490 NEXT I                                            <504>
2500 IF NOT F THEN PRINT "(7DOWN,3RIGHT)NICHT VOR
     HANDEN. KEINE LOESCHUNG !                         <PEV>
2510 GOTO 50                                           <3HB>
2520 PRINT "(2HOME,CLR,RVSON,4SPACE)S(3SPACE)O(3S
     PACE)R(3SPACE)T(3SPACE)I(3SPACE)E(3SPACE)R(3
     SPACE)E(3SPACE)N(3SPACE,RVOFF)"                   <ECH>
2530 PRINT "(2DOWN)SORTIEREN VON ";N;" BUCHUNGEN
     BEGINNT "                                         <44H>
2540 FOR I=2 TO N                                      <KNH>
2550 XY=VAL(MID$(P$(I),4,2))*100+VAL(MID$(P$(I),1
     ,2))                                              <FGV>
2560 VX=VAL(MID$(P$(I-1),4,2))*100+VAL(MID$(P$(I-
     1),1,2)): IF XY>=VX THEN 2650                     <FOD>
2570 V1$=U$(I): V2$=N$(I): V3$=S$(I): V4$=P$(I):
     V5$=T$(I)                                         <GL6>
2580 FOR Z=I-1 TO 1 STEP -1                            <TEA>
2590 U$(Z+1)=U$(Z): N$(Z+1)=N$(Z): S$(Z+1)=S$(Z):
      P$(Z+1)=P$(Z): T$(Z+1)=T$(Z)                     <D1I>
2600 TZ=VAL(MID$(V4$,4,2))*100+VAL(MID$(V4$,1,2))      <3BD>
2610 UV=VAL(MID$(P$(Z-1),4,2))*100+VAL(MID$(P$(Z-
     1),1,2)): IF TZ<=UV THEN 2640                     <VD9>
2620 U$(Z)=V1$: N$(Z)=V2$: S$(Z)=V3$: P$(Z)=V4$:
     T$(Z)=V5$                                         <K1M>
2630 GOTO 2650                                         <0BH>
2640 NEXT Z                                            <FTD>
2650 NEXT I                                            <5G4>
2660 PRINT "(2DOWN)SORTIEREN IM HAUPTSPEICHER BEE
     NDET."                                            <7RG>
2670 GOTO 50                                           <3HB>
2680 CHAR ,1,12,"BETRAG(10SPACE)[(9SPACE)]": LE$=
     "B": PU=7: GOSUB 2950                             <V4B>
2690 IF VAL(G$)>9999.99 THEN 2680                      <J81>
2700 IF VAL(G$)<0.01 THEN 2680                         <AGG>
2710 G1=VAL(G$): G$=RIGHT$(STR$(G1),LEN(STR$(G1))
     -1): G$=RIGHT$(L$+G$,7)                           <NB9>
2720 IF MID$(G$,5,1)="." THEN 2750                     <TTV>
2730 IF MID$(G$,6,1)="." THEN G$=G$+"0": G$=RIGHT
     $(G$,7): GOTO 2750                                <SKE>
2740 IF MID$(G$,7,1)>="0" AND MID$(G$,7,1)<="9" T
     HEN G$=G$+".00"                                   <SOD>
2750 G$=RIGHT$(G$,7)                                   <V0S>
2760 CHAR ,12,12,"(5SPACE)[ ": PRINT G$                <A6M>
2770 RETURN                                            <OU7>
2780 CHAR ,1,14,"DATUM(11SPACE)[(10SPACE)]": LE$=
     "D": PU=8: GOSUB 2960                             <JK9>
2790 PRINT "(UP,19RIGHT)";D$                           <N6N>
2800 IF VAL(MID$(D$,1,2))>31 OR VAL(MID$(D$,1,2))
     <1 THEN 2780                                      <SB1>
2810 IF VAL(MID$(D$,4,2))>12 OR VAL(MID$(D$,4,2))
     <1 THEN PRINT "(2UP)";: GOTO 2780                 <HVM>
2820 IF Q$="" OR LEN(Q$)<6 GOTO 2780                   <976>
2830 IF MID$(Q$,2,1)="." OR MID$(Q$,5,1)="." THEN
      2780                                             <DVR>
2840 IF MID$(Q$,4,1)="." OR MID$(Q$,7,1)="." OR M
     ID$(Q$,8,1)="." THEN 2780                         <N3J>
2850 RETURN                                            <P67>
2860 CHAR ,1,16,"ART +/- ?(7SPACE)[ ]                  <5A6>
2870 GET KEY A$: VOL 3: SOUND 1,15000,2                <HKT>
2880 IF A$="+" THEN PRINT "(3LEFT)[+]": M$="+": G
     OTO 2900                                          <CT2>
2890 IF A$="-" THEN PRINT "(3LEFT)[-]": M$="-": G
     OTO 2900: ELSE 2870                               <DJG>
2900 RETURN                                            <987>
2910 SCNCLR : PRINT : DIRECTORY : GOTO 50              <BE5>
2920 Q$="": CHAR ,1,8,"ABBUCHUNGSART(3SPACE)[(3SP
     ACE)](3LEFT)": LE$="V": PU=1: TG=31: HG=91:
     GOTO 2970                                         <PPD>
2930 Q$="": CHAR ,1,10,"BEGRIFF(9SPACE)[(20SPACE)
     ](20LEFT)": LE$="BE": PU=18                       <789>
2940 TG=31: HG=255: GOTO 2970                          <FL2>
2950 Q$="": CHAR ,1,12,"BETRAG(10SPACE)[(9SPACE)]
     (9LEFT)": TG=43: HG=57: GOTO 2970                 <40L>
2960 Q$="": CHAR ,1,14,"DATUM(11SPACE)[(10SPACE)]
     (10LEFT)": TG=45: HG=57: GOTO 2970                <B77>
2970 GET KEY A$: VOL 3: SOUND 1,15000,2: IF A$=CH
     R$(148) OR A$=CHR$(34) OR A$=CHR$(44) THEN 2
     970                                               <PPP>
2980 IF A$<>CHR$(20) AND A$<>CHR$(13) AND LEN(Q$)
     =PU THEN 2970                                     <HHE>
2990 IF A$=CHR$(157) OR A$=CHR$(17) OR A$=CHR$(14
     5) OR A$=CHR$(29) THEN 2970                       <LG9>
3000 IF A$=CHR$(20) AND LEN(Q$)<1 THEN 2970            <HN8>
3010 IF A$=CHR$(20) THEN PRINT CHR$(157);" ";CHR$
     (157);: Q$=LEFT$(Q$,LEN(Q$)-1)                    <N0P>
3020 IF A$=CHR$(13) AND LEN(Q$)>0 THEN 3080            <295>
3030 IF A$=CHR$(13) THEN 2970                          <350>
3040 IF ASC(A$)<TG OR ASC(A$)>HG THEN 2970             <9BG>
3050 PRINT A$;                                         <KQB>
3060 Q$=Q$+A$                                          <7KD>
3070 GOTO 2970                                         <4BD>
3080 PRINT                                             <B5U>
3090 IF LE$="V" THEN V$=Q$: GOTO 3140                  <KNE>
3100 IF LE$="BE" THEN B$=Q$: GOTO 3140                 <PTU>
3110 IF LE$="B" THEN G$=Q$: GOTO 3140                  <GTV>
3120 IF MID$(Q$,3,1)="." AND MID$(Q$,6,1)="." THE
     N D$=Q$: GOTO 3140                                <RH8>
3130 D$=MID$(Q$,1,2)+"."+MID$(Q$,3,2)+"."+MID$(Q$
     ,5,2): GOTO 3140                                  <ANS>
3140 RETURN                                            <887>
3150 SCNCLR : CHAR ,6,8,"<H> ILFE(2SPACE)ODER(2SP
     ACE)<E> NDE..?"                                   <2B1>
3160 GET KEY A$                                        <4GD>
3170 IF A$="H" THEN 3190                               <SHH>
3180 IF A$="E" THEN 3360: ELSE 3160                    <B64>
3190 PRINT "(2HOME,CLR,RVSON,7SPACE)P R O G R A M
      M H I L F E(8SPACE,RVOFF)": POKE 229,2           <9MI>
3200 PRINT "(DOWN)1. BEVOR NEUE BUCHUNGEN EINGEGE
     BEN                                               <8NE>
3210 PRINT "(3SPACE)WERDEN KOENNEN, MUSS DIE AKTU
     ELLE                                              <GKR>
3220 PRINT "(3SPACE)HAUSHALTSDATEI EINGELADEN WER
     DEN.                                              <GRO>
3230 PRINT "(3SPACE)ACHTUNG !! DER (RVSON)ALTE(RV
     OFF) FILENAME                                     <4JD>
3240 PRINT "(3SPACE)WIRD UEBERSCHRIEBEN."              <786>
3250 PRINT "(DOWN)2. BEI DER EINGABE GLEICHER BEG
     RIFFE                                             <FC9>
3260 PRINT "(3SPACE)SOLLTE DIE LFD. NUMMER MITANG
     EGEBEN                                            <3N0>
3270 PRINT "(3SPACE)WERDEN, DA SONST BEI DER SUCH
     ROUTINE                                           <S3B>
3280 PRINT "(3SPACE)SCHWIERIGKEITEN AUFTRETEN KOE
     NNEN.                                             <1J4>
3290 PRINT "(DOWN)3. DATUMSEINGABE -MIT-/ -OHNE-
     PUNKTE.                                           <PMV>
3300 PRINT "(DOWN)4. DIE DATEI WIRD NUR INNERHALB
      EINES                                            <621>
3310 PRINT "(3SPACE)JAHRES SORTIERT.(10SPACE)"         <UBV>
3320 PRINT "(DOWN)5. NACH EINEM PROGRAMMABBRUCH K
     ANN MAN                                           <6Q4>
3330 PRINT "(3SPACE)MIT ' GOTO 60' DAS PROGRAMM O
     HNE                                               <SPK>
3340 PRINT "(3SPACE)DATENVERLUST FORTSETZEN.           <NLQ>
3350 GOTO 50                                           <39B>
3360 PRINT "(12RIGHT,4DOWN,RVSON)ENDE(RVOFF)           <A7T>
3370 PRINT "(DOWN,4SPACE)MIT...GOTO 60...KANN DAS
      PROGRAMM"                                        <1RD>
3380 PRINT "(DOWN,4SPACE)OHNE DATENVERLUST FORTGE
     SETZT WERDEN": POKE 808,110: END                  <FD5>
3390 PRINT "(2HOME,CLR)": WINDOW 0,0,39,1: PRINT
     "(RVSON,4SPACE)M(2SPACE)O(2SPACE)T(2SPACE)I(
     2SPACE)Z(2SPACE)Z(2SPACE)E(2SPACE)T(2SPACE)T
     (2SPACE)E(2SPACE)L(3SPACE,RVOFF)"                 <SQ5>
3400 WINDOW 0,1,39,24: FL=1: F4$="": F3$="": PRIN
     T                                                 <8MI>
3410 P1$="R**********************S"                    <45C>
3420 P2$="W**********************W"                    <9RH>
3430 P3$="-(21SPACE)-"                                 <9T0>
3440 P4$="Z**********************X"                    <S3V>
3450 P5$="G E S A M T(SHF.SPACE) ----(SHF.SPACE)
     K(SHF.SPACE)A(SHF.SPACE)L(SHF.SPACE)K(SHF.SP
     ACE)U(SHF.SPACE)L(SHF.SPACE)A(SHF.SPACE)T(SH
     F.SPACE)I(SHF.SPACE)O(SHF.SPACE)N"                <ITK>
3460 M4$="----------------------------------------
     -"                                                <3HD>
```

```
3470 PRINT P1$: PRINT "R*>> AUSWAHL 1 - 6 <<*W":
     PRINT P2$: FOR U=1 TO 16: PRINT P3$: NEXT U:
      PRINT P2$: PRINT P3$: PRINT P4$                <DC3>
3480 PRINT "(16UP)- -1- ABBG. AUSWAEHLEN             <IRH>
3490 PRINT "-(DOWN) -2- ABBG. AUFLISTEN              <UNM>
3500 PRINT "-(DOWN) -3- GES.KALKULATION              <SRR>
3510 PRINT "-(DOWN) -4- TASCHENRECHNER               <8T6>
3520 PRINT "-(DOWN) -5- SUCHROUTINE                  <6AP>
3530 PRINT "-(DOWN) -6- M E(SHF.SPACE)N(SHF.SPACE
     )U(SHF.SPACE)E                                  <LD3>
3540 PRINT "-(3DOWN,2SPACE)BITTE WAEHLEN SIE         <2CT>
3550 GET KEY A$: A=VAL(A$): IF A<1 OR A>6 THEN 35
     50                                              <0P7>
3560 ON A GOTO 3570,3740,5280,4340,5610,60           <LLE>
3570 GOSUB 5620: PRINT "(CLR,HOME)BBBBBBBBBBBBBBBB
     BBBBBBBBBBBBBBBBBBBBBBBB"                       <71S>
3580 PRINT "R S-> SPEICHERN - U-> UEBERSPRINGEN(2
     SPACE)R"                                        <KGJ>
3590 PRINT "R M-> MENUE(5SPACE)- N-> NEUE EINFUEG
     EN R"                                           <P7S>
3600 PRINT "TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
     T"                                              <352>
3610 PRINT "(RVSON) -BEGRIFF(2SPACE)-(2SPACE)DM-
     -ART(RVOFF)": PRINT : POKE 229,7                <FHN>
3620 F4$=" < GESPEICHERT >": F3$=" < UEBERSPRINGE
     N >"                                            <R61>
3630 O=0: FOR I=1 TO 72                              <B5V>
3640 PRINT L1$(I);                                   <QPV>
3650 GET KEY A$: IF A$="S" THEN PRINT F4$: O=O+1:
      L2$(O)=L1$(I): GOTO 3690                       <7CF>
3660 IF A$="U" THEN PRINT F3$: GOTO 3690             <RNT>
3670 IF A$="M" THEN 3390                             <V9R>
3680 IF A$="N" THEN GOSUB 3710: O=O+1: L2$(O)=L1$
     (I): ELSE 3650                                  <06Q>
3690 NEXT I                                          <5O4>
3700 GOTO 3390                                       <1GJ>
3710 PRINT "(22LEFT) BITTE EINE NEUE ABBUCHUNG EI
     NFUEGEN(2SPACE)";                               <B98>
3720 INPUT L1$(I): PRINT                             <KAL>
3730 RETURN                                          <OU7>
3740 PRINT "(CLR,2HOME)"                             <PLQ>
3750 PRINT "(2HOME,2SPACE)BEGRIFF(5SPACE)DM(3SPAC
     E)ART(16SPACE)"                                 <QPN>
3760 PRINT " ----------------------"                 <IBO>
3770 PRINT CHR$(27)"T": PRINT "(22RIGHT,21DOWN)";
     CHR$(27)"B"                                     <DJN>
3780 RU=0: SCNCLR : FOR I=1 TO O: PRINT L2$(I)       <2BE>
3790 GOSUB 5250                                      <HVS>
3800 IF INT(I/13)<>I/13 THEN 3880                    <M2I>
3810 PRINT "(2DOWN)R**********************S"         <8LJ>
3820 PRINT "-<W> WEITER(2SPACE)<M> MENUE-"           <GL7>
3830 PRINT "-<T> TASCHENRECHNER(3SPACE)-"            <GO1>
3840 PRINT "Z**********************X"                <62E>
3850 GET KEY A$: IF A$="W" THEN SCNCLR : GOTO 388
     0                                               <BK0>
3860 IF A$="T" THEN 4360                             <QHL>
3870 IF A$="M" THEN 3390: ELSE 3850                  <2E5>
3880 NEXT I                                          <D54>
3890 IF RU>0 THEN PRINT "(DOWN) GUTHABEN(2SPACE)"
     :: PRINT USING "#####";RU                       <49B>
3900 IF RU<0 THEN PRINT "(DOWN) SOLL(5SPACE)";: P
     RINT USING "-#####";RU                          <J5B>
3910 IF RU=0 THEN PRINT "(DOWN) SUMME +/-(2SPACE)
     ";: PRINT USING "####";RU                       <MSP>
3920 PRINT "(DOWN)R**********************S"          <5RP>
3930 PRINT "-<A> ANFANG(2SPACE)<M> MENUE-"           <DV2>
3940 PRINT "-<D> ABBUCHUNG DRUCKEN-"                 <M95>
3950 PRINT "-<T> TASCHENRECHNER(3SPACE)-"            <GG0>
3960 PRINT "Z**********************X"                <62E>
3970 GET KEY A$: IF A$="A" THEN SCNCLR : GOTO 377
     0                                               <3P1>
3980 IF A$="T" THEN 4360                             <PHP>
3990 IF A$="D" THEN 4010                             <THP>
4000 IF A$="M" THEN 3390: ELSE 3970                  <7U5>
4010 OPEN 1,4,7: CMD 1: PRINT TAB(16)" BEGRIFF(5S
     PACE)DM(3SPACE)ART"                             <MQJ>
4020 PRINT TAB(16)" TTTTTTTTTTTTTTTTTTTTTT": PRINT
     : RU=0: FOR I=1 TO O                            <DF5>
4030 GOSUB 5250: PRINT TAB(15)L2$(I)                 <NUT>
4040 NEXT I                                          <D54>
4050 PRINT TAB(16)" TTTTTTTTTTTTTTTTTTTTTT": PRINT   <BPJ>
4060 IF RU>0 THEN PRINT TAB(14)" GUTHABEN(3SPACE)
     ";: PRINT USING "#######";RU: CLOSE 1: GOTO
     3970                                            <A3N>
4070 IF RU<0 THEN PRINT TAB(13)" SOLL(7SPACE)";:
     PRINT USING "-#######";RU: CLOSE 1: GOTO 397
     0                                               <A1Q>
4080 IF RU=0 THEN PRINT TAB(14)" SUMME +/-(2SPACE
     )";: PRINT USING "#####";RU: CLOSE 1: GOTO 3
     970                                             <T7Q>
4090 PU=11                                           <DH7>
4100 PRINT "(2HOME,RVSON,4SPACE)N(2SPACE)O(2SPACE
     )T(2SPACE)I(2SPACE)Z(2SPACE)Z(2SPACE)E(2SPAC
     E)T(2SPACE)T(2SPACE)E(2SPACE)L(3SPACE,RVOFF)
     "                                               <15M>
4110 PRINT "(22SPACE)"                               <C0U>
4120 CHAR ,0,2,"R**********************S"            <DL1>
4130 CHAR ,0,3,"R*>> BESCHREIBUNG(2SPACE)<<*W"       <MIF>
4140 CHAR ,0,4,"R**********************W"            <UB4>
4150 CHAR ,0,5,"-(21SPACE)-"                         <MTH>
4160 CHAR ,0,6,"- 4 GRUNDRECHENARTEN(2SPACE)-"       <CBK>
4170 CHAR ,0,7,"-(21SPACE)-"                         <LTP>
4180 CHAR ,0,8,"- <INFO> INFORMATION(2SPACE)-"       <CT0>
4190 CHAR ,0,9,"- <Z> HAUPTMENUE(6SPACE)-"           <C17>
4200 CHAR ,0,10,"- <A> NEUSTART(8SPACE)-"            <B6K>
4210 CHAR ,0,11,"- <O> NOTIZZETTEL(5SPACE)-"         <Q9A>
4220 CHAR ,0,12,"- <C> EINGABE LOESCHEN-"            <J64>
4230 CHAR ,0,13,"- <H> ZEIGT HABEN/SOLL-"            <9UK>
4240 CHAR ,0,14,"- <P> ABBG. AUFLISTEN -"            <543>
4250 CHAR ,0,15,"- <L> SPEICHER LOESCH.-"            <3VH>
4260 CHAR ,0,16,"- <%> PROZENTRECHNUNG -"            <HHB>
4270 CHAR ,0,17,"- <S> SPEICHERN(7SPACE)-"           <NFD>
4280 CHAR ,0,18,"- <M> SPEICHERAUFRUF(2SPACE)-"      <JA5>
4290 CHAR ,0,19,"- <K> KALKULAT.ERGEBN.-"            <F6A>
4300 CHAR ,0,20,"-(21SPACE)-"                        <01U>
4310 CHAR ,0,21,"R**********************W"           <BN7>
4320 CHAR ,0,23,"Z**********************X"           <FBH>
4330 CHAR ,0,22,"-(RVSON,3SPACE)RECHNER(3SPACE)AK
     TIV(3SPACE,RVOFF)-": GOTO 4350                  <S72>
4340 CHAR ,0,21,"-(RVSON,3SPACE)RECHNER(3SPACE)AK
     TIV(3SPACE,RVOFF)-"                             <G18>
4350 PRINT "(2HOME)": PU=11: CHAR ,23,2,"(RVSON)R
     **************S": GOTO 4380                     <MM9>
4360 PRINT "(2HOME)": PU=11: CHAR ,24,0,"(RVSON)R
     ECHNER(2SPACE)AKTIV"                            <F0F>
4370 CHAR ,23,2,"(RVSON)R**************S"            <LK1>
4380 CHAR ,23,3,"-TASCHENRECHNER-"                   <5UN>
4390 CHAR ,23,4,"-(2SPACE)BY M.WEBER(2SPACE)-"       <30R>
4400 CHAR ,23,5,"-R************S-"                   <3RK>
4410 CHAR ,23,6,"--(RVOFF,12SPACE,RVSON)--"          <9FK>
4420 CHAR ,23,7,"--(RVOFF)>": CHAR ,26,7,E$: CHAR
      ,36,7,"<(RVSON)--": IF LEN(E$)>10 THEN GOSU
     B 7110                                          <7AL>
4430 CHAR ,23,8,"(RVSON)--(RVOFF,12SPACE,RVSON)--
     "                                               <OM1>
4440 CHAR ,23,9,"-Z************X-"                   <7J5>
4450 CHAR ,23,10,"-R*R*R*R*R*R**S-"                  <CB7>
4460 CHAR ,23,11,"--C-A-M-L-H-(RVOFF,2SPACE,RVSON
     )--"                                            <O6N>
4470 CHAR ,23,12,"-R*+*+*+*+*+**W-"                  <4O4>
4480 CHAR ,23,13,"--K-%-P-S-Z-(RVOFF,2SPACE,RVSON
     )--"                                            <47O>
4490 CHAR ,23,14,"-Z*E*E*E*E*E**X-"                  <3H7>
4500 CHAR ,23,15,"-R*R*R*R*R*****S-"                 <OMJ>
4510 CHAR ,23,16,"--7-8-9-*-(4SPACE)--"              <9FT>
4520 CHAR ,23,17,"-R*+*+*+*WINFO--"                  <GPG>
4530 CHAR ,23,18,"--4-5-6-/-(4SPACE)--"              <IA5>
4540 CHAR ,23,19,"-R*+*+*+*+****W-"                  <BKO>
4550 CHAR ,23,20,"--1-2-3-+-(4SPACE)--"              <8BT>
4560 CHAR ,23,21,"-R*+*+*+*WOFF --"                  <FIC>
4570 CHAR ,23,22,"--0-.-=---(4SPACE)--"              <DNL>
4580 CHAR ,23,23,"-Z*E*E*E*E*****X-": CHAR ,23,24,
     "Z**************X(RVOFF)"                       <C3P>
4590 GET KEY A$: VOL 3: SOUND 1,15000,2: IF A$=""
      THEN 4590                                      <DT7>
4600 IF A$="*" THEN 4780                             <51L>
4610 IF A$="/" THEN 4870                             <095>
4620 IF A$="+" THEN 4960: ELSE IF A$="=" THEN 514
     0                                               <7LC>
4630 IF A$="M" THEN E$="": E$=MID$(RU$,1,10): CHA
     R ,26,7,E$: GOTO 4350                           <MC7>
4640 IF A$="S" THEN RU$=MID$(RE$,1,10): E$=""        <CBP>
4650 IF A$="-" THEN 5050: ELSE IF A$="L" THEN E$=
     "": RU$="": RE$="": CHAR ,26,7,MP$: GOTO 435
     0                                               <049>
4660 IF A$="%" THEN F1=1: GOTO 5140                  <EIT>
4670 IF A$="Z" THEN 60                               <KA5>
4680 IF A$="I" THEN E$="": GOTO 4090                 <QS6>
4690 IF A$="H" THEN CHAR ,26,7,MP$: GOSUB 7010: S
     U=INT(SU*100+0.5)/100: E$=STR$(SU): CHAR ,26
     ,7,E$: RE$=E$: GOTO 4350                        <81F>
4700 IF A$="K" THEN CHAR ,26,7,MP$: GOSUB 7060: R
     U=INT(RU*100+0.5)/100: E$=STR$(RU): CHAR ,26
     ,7,E$: RE$=E$: GOTO 4350                        <HPG>
4710 IF A$="C" THEN E$="": CHAR ,26,7,MP$            <2TO>
4720 IF A$="A" THEN RE=0: RO=0: E$="": CHAR ,26,7
     ,MP$: GOTO 4350                                 <NSO>
4730 IF A$="P" THEN RE=0: RO=0: CHAR ,26,7,MP$: G
     OTO 3750                                        <UOT>
4740 IF A$="O" THEN RE=0: RO=0: CHAR ,26,7,MP$: G
     OTO 3390                                        <VN9>
4750 IF ASC(A$)<37 OR ASC(A$)>57 THEN 4350           <24T>
4760 IF A$=CHR$(148) OR A$=CHR$(34) OR A$=CHR$(44
     ) OR A$=CHR$(13) THEN 4350                      <0R9>
4770 E$=E$+A$: GOTO 4350                             <KC8>
4780 CHAR ,26,7,MP$                                  <TQ0>
4790 IF RO=0 THEN I=1: RO=1: E1(1)=VAL(E$): E$=""
     : GOTO 4350                                     <SJF>
4800 I=I+1: E1(I)=VAL(E$): E$=""                     <KND>
4810 IF RO=1 THEN RE=E1(I-1)*E1(I): I=1: E1(I)=RE    <RM2>
4820 IF RO=2 THEN RE=E1(I-1)/E1(I): I=1: E1(I)=RE    <PIB>
4830 IF RO=3 THEN E1(0)=E1(1): FL=1                  <R6P>
4840 IF RO=4 THEN E1(0)=E1(1): FL=-1                 <5A6>
4850 RO=1                                            <AL3>
4860 GOTO 4350                                       <4G3>
4870 CHAR ,26,7,MP$                                  <LA2>
4880 IF RO=0 THEN I=1: RO=2: E1(1)=VAL(E$): E$=""
     : GOTO 4350                                     <JF0>
4890 I=I+1: E1(I)=VAL(E$): E$=""                     <G7O>
4900 IF RO=1 THEN RE=E1(I-1)*E1(I): I=1: E1(I)=RE    <RQ2>
4910 IF RO=2 THEN RE=E1(I-1)/E1(I): I=1: E1(I)=RE    <PMB>
4920 IF RO=3 THEN E1(0)=E1(1): FL=1                  <O6P>
4930 IF RO=4 THEN E1(0)=E1(1): FL=-1                 <FA6>
4940 RO=2                                            <6OI>
4950 GOTO 4350                                       <4O3>
4960 CHAR ,26,7,MP$                                  <DA2>
4970 IF RO=0 THEN I=1: RO=3: E1(1)=VAL(E$): E$=""
     : GOTO 4350                                     <MR6>
4980 I=I+1: E1(I)=VAL(E$): E$=""                     <HND>
4990 IF RO=1 THEN RE=E1(I-1)*E1(I)*FL+E1(0)          <A3C>
5000 IF RO=2 THEN RE=E1(I-1)/E1(I)*FL+E1(0)          <G0K>
5010 IF RO=3 THEN RE=E1(I-1)+E1(I)                   <FL4>
5020 IF RO=4 THEN RE=E1(I-1)-E1(I)                   <QNS>
```

Listing 1. »HAUSHALT« (Fortsetzung)

```
5030 RO=3: I=1: E1(I)=RE                                   <0G6>
5040 GOTO 4350                                             <103>
5050 CHAR ,26,7,MP$                                        <5U0>
5060 IF RO=0 THEN I=1: RO=4: E1(1)=VAL(E$): E$=""
     : GOTO 4350                                           <87D>
5070 I=I+1: E1(I)=VAL(E$): E$=""                           <T7D>
5080 IF RO=1 THEN RE=E1(I-1)*E1(I)*FL+E1(0)                <46K>
5090 IF RO=2 THEN RE=E1(I-1)/E1(I)*FL+E1(0)                <22S>
5100 IF RO=3 THEN RE=E1(I-1)+E1(I)                         <CL4>
5110 IF RO=4 THEN RE=E1(I-1)-E1(I)                         <RNS>
5120 RO=4: I=1: E1(I)=RE                                   <500>
5130 GOTO 4350                                             <0B3>
5140 CHAR ,26,7,MP$                                        <TU0>
5150 I=I+1: E1(I)=VAL(E$)                                  <4LK>
5160 IF RO=1 THEN RE=E1(I-1)*E1(I)*FL+E1(0)                <S5K>
5170 IF RO=2 THEN RE=E1(I-1)/E1(I)*FL+E1(0)                <Q1S>
5180 IF RO=3 THEN RE=E1(I-1)+E1(I)                         <4L4>
5190 IF RO=4 THEN RE=E1(I-1)-E1(I)                         <JNS>
5200 I=0: E$=""                                            <34G>
5210 IF F1=1 THEN RE=RE*100: F1=0                          <MAU>
5220 RE$=STR$(RE): LU=LEN(RE$): IF LU>PU THEN RI$
     =MID$(RE$,1,10): GOTO 5230: ELSE 5240                 <TTV>
5230 RE$=MID$(RI$,1,10)                                    <7BB>
5240 CHAR ,26,7,RE$: RE=0: RO=0: E$="": E1(0)=0:
     E1(1)=0: E1(2)=0: E1(3)=0: GOTO 4350                  <USU>
5250 R2$=MID$(L2$(I),13,4): R1$=MID$(L2$(I),1,1)           <VQ2>
5260 IF R1$="-" THEN RU=RU-VAL(R2$): ELSE RU=RU+V
     AL(R2$)                                               <H4C>
5270 RETURN                                                <PI7>
5280 PRINT "{CLR,4DOWN}": PRINT TAB(8)P1$: PRINT
     TAB(8)"={2SPACE}-B-{3SPACE}BILDSCHIRM{3SPACE
     }="                                                   <INA>
5290 PRINT TAB(8)P2$: PRINT TAB(8)"={2SPACE}-D-{3
     SPACE}DRUCKER{6SPACE}=": PRINT TAB(8)P4$              <206>
5300 Y=0: GET KEY A$: IF A$="B" THEN PRINT "{2HOM
     E,CLR}": GOSUB 5330: GOSUB 6980: GOTO 3390            <FEP>
5310 IF A$="D" THEN GOSUB 6990: GOSUB 5320: GOSUB
      6980: GOTO 3390: ELSE 5300                           <GD2>
5320 Y=4: OPEN 1,4,7: CMD 1: PRINT P5$: GOTO 5340          <EOC>
5330 PRINT P5$                                             <TVN>
5340 PRINT DA$: PRINT TI$;: PRINT " UHR{10SPACE}B
     EGRIFF{5SPACE}DM{3SPACE}ART": PRINT M4$: POK
     E 229,5                                               <03L>
5350 RU=0: FOR I=1 TO O: IF O=0 THEN 5360: ELSE 5
     370                                                   <NIA>
5360 PRINT "{3SPACE}KEINE DATEN VORHANDEN !!{2DOW
     N}": CLOSE 1: FOR U=1 TO 1000: NEXT U: GOSUB
      6980: GOTO 3390                                      <BJF>
5370 PRINT "{18SPACE}";L2$(I): R2$=MID$(L2$(I),13
     ,4): R1$=MID$(L2$(I),1,1)                             <G2A>
5380 IF R1$="-" THEN RU=RU-VAL(R2$): ELSE RU=RU+V
     AL(R2$)                                               <HSF>
5390 IF Y=4 THEN 5460: ELSE 5400                           <THJ>
5400 IF INT(I/12)<>I/12 THEN 5460                          <5MS>
5410 PRINT "{2DOWN}": PRINT L6$: PRINT "= AUSWAHL
      W-WEITER{2SPACE}A-ANFANG{2SPACE}*-MENUE{2SP
     ACE}=": PRINT L7$                                     <8OU>
5420 GET KEY A$                                            <10D>
5430 IF A$="W" THEN SCNCLR : GOTO 5460                     <H72>
5440 IF A$="*" THEN 3390                                   <194>
5450 IF A$="A" THEN SCNCLR : GOTO 5350: ELSE 5420          <NN3>
5460 NEXT I: PRINT M4$                                     <8NF>
5470 IF RU>0 THEN 5480: ELSE 5490                          <19D>
5480 PRINT "= KALKULATIONS ( GUTHABEN ){2SPACE}";
     : PRINT USING "#####.##";RU;: PRINT D$: GOTO
      5510                                                 <P6F>
5490 IF RU<0 THEN 5500                                     <AKM>
5500 PRINT "= KALKULATIONS ( SOLL ){5SPACE}";: PR
     INT USING "-#####.##";RU;: PRINT D$                   <5OQ>
5510 GOSUB 7010: IF SU>0 THEN 5520: ELSE 5530              <DDB>
5520 PRINT "+ AKTUELLES{4SPACE}( GUTHABEN ){2SPAC
     E}";: PRINT USING "#####.##";SU;: PRINT D$:
     GOTO 5550                                             <MPM>
5530 IF SU<0 THEN 5540                                     <A4G>
5540 PRINT "- AKTUELLES{4SPACE}( SOLL ){5SPACE}";
     : PRINT USING "-#####.##";SU;: PRINT D$               <I3O>
5550 PRINT M4$                                             <HMC>
5560 RA=RU+SU: IF RA>0 THEN 5570: ELSE 5580                <777>
5570 PRINT "= GESAMTGUTHABEN{12SPACE}";: PRINT US
     ING "-#####.##";RA;: PRINT D$: GOTO 5600              <44I>
5580 IF RA<0 THEN 5590                                     <IOV>
5590 PRINT "= GESAMTSOLL{16SPACE}";: PRINT USING
     "-#####.##";RA;: PRINT D$                             <UHI>
5600 PRINT M4$: CLOSE 1: RETURN                            <72K>
5610 GOTO 1620                                             <50Q>
5620 RESTORE 5670                                          <E7R>
5630 FOR I=1 TO 72                                         <ACQ>
5640 READ L1$(I)                                           <AGN>
5650 NEXT I                                                <5C4>
5660 RETURN                                                <907>
5670 DATA "- BEGRIFF.. 0000{3SPACE}MTL"                    <0CF>
5680 DATA "- BEGRIFF.. 0000{3SPACE}MTL"                    <S8F>
5690 DATA "- BEGRIFF.. 0000{3SPACE}MTL"                    <O8F>
5700 DATA "- BEGRIFF.. 0000{3SPACE}MTL"                    <KCF>
5710 DATA "- BEGRIFF.. 0000{3SPACE}MTL"                    <GCF>
5720 DATA "- BEGRIFF.. 0000{3SPACE}MTL"                    <SCF>
5730 DATA "+ BEGRIFF.. 0000{3SPACE}MTL"                    <8AF>
5740 DATA "+ BEGRIFF.. 0000{3SPACE}MTL"                    <KAF>
5750 DATA "+ BEGRIFF.. 0000{3SPACE}MTL"                    <GAF>
5760 DATA "+ BEGRIFF.. 0000{3SPACE}MTL"                    <SAF>
5770 DATA "+ BEGRIFF.. 0000{3SPACE}MTL"                    <OAF>
5780 DATA "+ BEGRIFF.. 0000{3SPACE}MTL"                    <4EF>
5790 DATA " {RVSON}.....J A N U A R.....{RVOFF}"           <TF8>
5800 DATA "- BEGRIFF.. 0000 * JAN"                         <94Q>
5810 DATA "- BEGRIFF.. 0000 * JAN"                         <D4Q>
```

Listing 1. »HAUSHALT« (Fortsetzung)



Sonderhefte im Überblick

Die 64er-Sonderhefte bieten Ihnen detaillierte Informationen zu speziellen Themen rund um die Commodore-Computer.

Bestellen Sie bitte die gewünschten Sonderhefte zum Preis von jeweils 14,– DM mit der beigehefteten Zahlkarte nach Seite 34.

Grafik

Drucker



SONDERHEFT 0018: DRUCKER
Listing: professionelle Textverarbeitung für den MPS 801 / Matrixdrucker im Test



SONDERHEFT 0032: FLOPPYLAUFWERKE UND DRUCKER
Tips & Tools, RAM-Erweiterung des C64, Druckerroutinen



SONDERHEFT 0020: GRAFIK
Grafik-Programmierung / Bewegungen



SONDERHEFT 0023: GRAFIK / ANWENDUNGEN
Paint Magic: ein professionelles Malprogramm



SONDERHEFT 0027 GRAFIK
AMICA Paint: Malprogramm



SONDERHEFT 0034: GRAFIK, SIMULATION, LERNEN
Konstruieren mit dem C64, Kurvendiskussion, Einstieg in Digitaltechnik

C 128

C 64-Einstieg



SONDERHEFT 0005: C 64-GRUNDWISSEN
Vom ersten Einschalten bis zum eigenen Programm / Grundlagen, Tips und Tricks



SONDERHEFT 0016: EINSTEIGER 2
Spriteanimation: Zeichentrickfilm mit dem Computer / GEOS, die neue Benutzeroberfläche



SONDERHEFT 0026 RUND UM DEN C64
Der C64 verständlich für alle mit ausführlichen Kursen.



SONDERHEFT 0019: EINSTEIGER 3
Basic-Kurs / Programm-Übersicht



SONDERHEFT 0001: C 128
Das können C 128 und C 128 D / Vergleich: C 128–C 64 / die passende Peripherie



SONDERHEFT 0010: C 128 II
Die Geheimnisse von CP/M / Kompletter C 128-Schaltplan / Grafik für Einsteiger



SONDERHEFT 0022: C 128 III
Farbiges Scrolling im 80-Zeichen-Modus / 8-Sekunden-Kopierprogramm



SONDERHEFT 0029 C128
Starke Software für C128/128D. Alles über den neuen C128D im Blechgehäuse.

Tips & Tricks, Anwendungen

Spiele



SONDERHEFT 9903: SPIELE
Top-Spiele-Listings für C 64 und VC 20 / Große Spiele-Marktübersicht



SONDERHEFT 0017 SPIELE FÜR C 64 UND C 128
So programmiert man Scrolling / Strategiespiele: Grips ist gefragt



SONDERHEFT 0030 SPIELE FÜR C64 und C128
Tolle Spiele zum Abtippen für C64/128. Spieleprogrammierung.



SONDERHEFT 9901: TIPS & TRICKS
Befehlserweiterungen für Betriebssystem und Floppy / Unentbehrliche Programmierhilfen




SONDERHEFT 0002: TIPS & TRICKS
Zeichensatz- und Sprite-Editor / Interrupt-Joystickabfrage / 27 nützliche Einzeiler



SONDERHEFT 0024: TIPS, TRICKS & TOOLS
Automatische Textkorrektur / Utilities / Basic-Compiler zum Abtippen



SONDERHEFT 0033: TIPS, TRICKS & TOOLS
Basic-Control-System, Titelgenerator, Digitale Super-Sounds, Betriebssysteme im Vergleich



SONDERHEFT 0031: DFÜ, MUSIK, MESSEN – STEUERN – REGELN
Alles über DFÜ, BTX von A-Z, Grundlagen, Bauanleitungen, C64 als Soundgigant

Floppy, Datasette, Dateiverwaltung

Programmiersprachen



SONDERHEFT 0012: PROGRAMMIERSPRACHEN
Pascal, Comal, Prolog, C und Forth / Vergleich: Basic-Compiler



SONDERHEFT 0021: ASSEMBLER UND BASIC
Giga-Ass: Hypra-Ass hoch 2 / Paradoxon-Basic: 50000 Basic Bytes free



SONDERHEFT 0035: ASSEMBLER
Abgeschlossene Kurse für Anfänger und Fortgeschrittene



SONDERHEFT 0025: FLOPPY / DATASETTE
Kurse: Floppy-Programmierung für Einsteiger und Profis



SONDERHEFT 0009: FLOPPY & DATEIVERWALTUNG
Floppy-Beschleuniger im Vergleichstest / Arbeiten mit dBase II / C 128-Diskmonitor



SONDERHEFT 0015: FLOPPY & DATASETTE
Reparaturanleitung: Erste Hilfe für die Diskettenstation / Hypratape: das Super-Turbotape



SONDERHEFT 0028 GEOS, DATEIVERWALTUNG
Viele Kurse zu Geos. Tolle Geos-Programme zum Abtippen.





SONDERHEFT 0008 PLUS/4 UND C 16
Übersicht: Zeropage und wichtige Systemadressen / Grundlagen und viele Listings

```
5820 DATA "+ BEGRIFF.. 0000 * JAN"                      <H2Q>
5830 DATA "+ BEGRIFF.. 0000 * JAN"                      <L2Q>
5840 DATA " {RVSON}....F E B R U A R....{RVOFF}"        <93K>
5850 DATA "- BEGRIFF.. 0000 * FEB"                      <G5E>
5860 DATA "- BEGRIFF.. 0000 * FEB"                      <S5E>
5870 DATA "+ BEGRIFF.. 0000 * FEB"                      <83E>
5880 DATA "+ BEGRIFF.. 0000 * FEB"                      <K7E>
5890 DATA " {RVSON}....M{2SPACE}A{2SPACE}E{2SPACE
     }R{2SPACE}Z....{RVOFF}"                            <HI7>
5900 DATA "- BEGRIFF.. 0000 * MAE"                      <4SD>
5910 DATA "- BEGRIFF.. 0000 * MAE"                      <0SD>
5920 DATA "+ BEGRIFF.. 0000 * MAE"                      <SQD>
5930 DATA "+ BEGRIFF.. 0000 * MAE"                      <OQD>
5940 DATA " {RVSON}....A{2SPACE}P{2SPACE}R{2SPACE
     }I{2SPACE}L....{RVOFF}"                            <E40>
5950 DATA "- BEGRIFF.. 0000 * APR"                      <HK8>
5960 DATA "- BEGRIFF.. 0000 * APR"                      <TK8>
5970 DATA "+ BEGRIFF.. 0000 * APR"                      <9I8>
5980 DATA "+ BEGRIFF.. 0000 * APR"                      <LM8>
5990 DATA " {RVSON}.......M{2SPACE}A{2SPACE}I....
     ...{RVOFF}"                                        <6HB>
6000 DATA "- BEGRIFF.. 0000 * MAI"                      <VG0>
6010 DATA "- BEGRIFF.. 0000 * MAI"                      <RG0>
6020 DATA "+ BEGRIFF.. 0000 * MAI"                      <7M0>
6030 DATA "+ BEGRIFF.. 0000 * MAI"                      <3M0>
6040 DATA " {RVSON}.....J{2SPACE}U{2SPACE}N{2SPAC
     E}I......{RVOFF}"                                  <07R>
6050 DATA "- BEGRIFF.. 0000 * JUN"                      <79J>
6060 DATA "- BEGRIFF.. 0000 * JUN"                      <B9J>
6070 DATA "+ BEGRIFF.. 0000 * JUN"                      <VFJ>
6080 DATA "+ BEGRIFF.. 0000 * JUN"                      <3BJ>
6090 DATA " {RVSON}.....J{2SPACE}U{2SPACE}L{2SPAC
     E}I......{RVOFF}"                                  <IVO>
6100 DATA "- BEGRIFF.. 0000 * JUL"                      <RPH>
6110 DATA "- BEGRIFF.. 0000 * JUL"                      <VPH>
6120 DATA "+ BEGRIFF.. 0000 * JUL"                      <3VH>
6130 DATA "+ BEGRIFF.. 0000 * JUL"                      <7VH>
6140 DATA " {RVSON}..A{2SPACE}U{2SPACE}G{2SPACE}U
     {2SPACE}S{2SPACE}T...{RVOFF}"                      <0IQ>
6150 DATA "- BEGRIFF.. 0000 * AUG"                      <60M>
6160 DATA "- BEGRIFF.. 0000 * AUG"                      <A0M>
6170 DATA "+ BEGRIFF.. 0000 * AUG"                      <U6M>
6180 DATA "+ BEGRIFF.. 0000 * AUG"                      <22M>
6190 DATA " {RVSON}..S E P T E M B E R..{RVOFF}"        <46E>
6200 DATA "- BEGRIFF.. 0000 * SEP"                      <LR9>
6210 DATA "- BEGRIFF.. 0000 * SEP"                      <HR9>
6220 DATA "+ BEGRIFF.. 0000 * SEP"                      <DT9>
6230 DATA "+ BEGRIFF.. 0000 * SEP"                      <9T9>
6240 DATA " {RVSON}.O{2SPACE}K{2SPACE}T{2SPACE}O{
     2SPACE}B{2SPACE}E{2SPACE}R.{RVOFF}"                <NSJ>
6250 DATA "- BEGRIFF.. 0000 * OKT"                      <RO1>
6260 DATA "- BEGRIFF.. 0000 * OKT"                      <NO1>
6270 DATA "+ BEGRIFF.. 0000 * OKT"                      <3U1>
6280 DATA "+ BEGRIFF.. 0000 * OKT"                      <VQ1>
6290 DATA " {RVSON}...N O V E M B E R...{RVOFF}"        <ETL>
6300 DATA "- BEGRIFF.. 0000 * NOV"                      <48F>
6310 DATA "- BEGRIFF.. 0000 * NOV"                      <08F>
6320 DATA "+ BEGRIFF.. 0000 * NOV"                      <SEF>
6330 DATA "+ BEGRIFF.. 0000 * NOV"                      <OEF>
6340 DATA " {RVSON}...D E Z E M B E R...{RVOFF}"        <MBN>
6350 DATA "- BEGRIFF.. 0000 * DEZ"                      <S9V>
6360 DATA "- BEGRIFF.. 0000 * DEZ"                      <G9V>
6370 DATA "+ BEGRIFF.. 0000 * DEZ"                      <4FV>
6380 DATA "+ BEGRIFF.. 0000 * DEZ"                      <OBV>
6390 PRINT "{2HOME,CLR,RVSON}F{2SPACE}I{2SPACE}L{
     2SPACE}E{2SPACE}S{4SPACE}L{2SPACE}O{2SPACE}E
     {2SPACE}S{2SPACE}C{2SPACE}H{2SPACE}E{2SPACE}
     N {RVOFF}": POKE 229,1                             <B3C>
6400 SCNCLR : PRINT : DIRECTORY                         <LK9>
6410 PRINT "{2DOWN}BITTE DEN NAMEN DES ZU LOESCHE
     NDEN FILES                                         <LM3>
6420 INPUT "EINGEBEN (*=MENUE) ";Q1$                    <BNR>
6430 IF Q1$="*" THEN 60: ELSE 6440                      <JS4>
6440 PRINT "{2DOWN}FILE " CHR$(34)Q1$ CHR$(34)" W
     IRD GELOESCHT": SCRATCH ""+Q1$                     <FLA>
6450 GOTO 60                                            <258>
6460 PRINT "{2HOME,CLR,RVSON,5SPACE}F I L E S{3SP
     ACE}U M B E N E N N E N{4SPACE,RVOFF}": POKE
      229,1                                             <L6J>
6470 SCNCLR : PRINT : DIRECTORY                         <5Q9>
6480 INPUT "{2DOWN}ALTER FILENAME (*=MENUE) ";Q1$
     : IF Q1$="*" THEN 60: ELSE 6490                    <SDB>
6490 INPUT "{DOWN}NEUER FILENAME";F1$: IF F1$=""
     GOTO 6490: ELSE K1$=CHR$(34)                       <CAJ>
6500 PRINT "{DOWN,2SPACE}FILE " CHR$(34)Q1$ CHR$(
     34)" WIRD AUF DEN": PRINT                          <J96>
6510 PRINT "{2SPACE}NAMEN " CHR$(34)F1$ CHR$(34)"
      UMBENANNT": RENAME ""+Q1$ TO ""+F1$: PRINT
     : PRINT                                            <MRV>
6520 GOTO 50                                            <INB>
6530 PRINT "{2HOME,CLR,RVSON} D I S K E T T E{4SP
     ACE}A U F R A E U M E N {RVOFF}": POKE 229,1       <O22>
6540 PRINT "{4DOWN,2RIGHT}BIST DU SICHER (J/N) ":
      GET KEY A$: IF A$="J" THEN 6560                   <R5P>
6550 IF A$="N" THEN 60: ELSE 6540                       <QN4>
6560 COLLECT                                            <OH3>
```

Schachverwal

Schachfreunde aufgepaßt, hier kommt ein geniales Verwaltungsprogramm für Ihre Turniere. Das Programm übernimmt die Auslosung der Paarungen nach dem Schweizer System und die Ergebnisverwaltung. Im Freundeskreis oder bei einem Turnier mit über 200 Spielern.

Freunde des königlichen Spiels kennen das Problem: Die Ansetzung der Paarungen und die Auswertung aller Ergebnisse ist enorm zeitaufwendig, Fehler fast schon vorhersehbar. Wenn dann noch nach dem »Schweizer System« (siehe Textkasten) gespielt wird, ergrauen selbst ausgefuchste Organisatoren in kurzer Zeit.

Was kann ein guter Organisator eines Turniers also diesen bekannten Schwierigkeiten entgegenstellen? Vor dieser Frage stand auch der Autor der »Schachverwaltung« (Listings 1 bis 6). Die Geschichte des Programms finden Sie im Textkasten auf Seite 22. Die Lösung war ein schnelles, durch Maschinenroutinen unterstütztes Basic-Programm auf dem C 128.

Das Programm hilft Ihnen bei der Gestaltung eines Turnieres mit maximal 254 Spielern. Die Ansetzung der Paarungen nimmt das Programm nach dem Schweizer System vor. Dies führt zu großer Gerechtigkeit und zur exzellenten Bewertung der Spieler schon nach wenigen Runden. Die Verwaltung wurde in der Zwischenzeit bei mehreren großen Turnieren eingesetzt, die Praxistauglichkeit ist längst bewiesen. Der große Komfort der Schachverwaltung wird durch eine ausgezeichnete Benutzerführung in den zahlreichen Menüs unterstützt. Natürlich sind deutsche Umlaute zugelassen. In jeder Phase eines Turniers sind die einzelnen Ergebnisse jeder bisher gespielten Runde anzuzeigen und sogar zu ändern. Spieler können im Turnierverlauf zurückgezogen werden, die Spieler- und Ergebnislisten sind auf einem Drucker auszugeben. Eine Centronics-Schnittstelle ist in das Programm integriert, der Ausdruck erfolgt über das Verbindungskabel am User-Port, also parallel. Jeder Drucker mit Centronics-Schnittstelle ist so anzusprechen. Wenn Sie mit einem Interface seriell ausdrucken wollen, ist das Programm vorher entsprechend zu ändern.

Nach dem Eingeben der Listings und anschließendem Speichern auf Diskette starten Sie das Ladeprogramm mit RUN"LADER"

Das Titelbild der Schachverwaltung wird aufgebaut und Sie werden aufgefordert, eine beliebige Taste zu drücken. Ist dies geschehen, werden nacheinander das Hauptprogramm und die Maschinenroutinen geladen. Anschließend

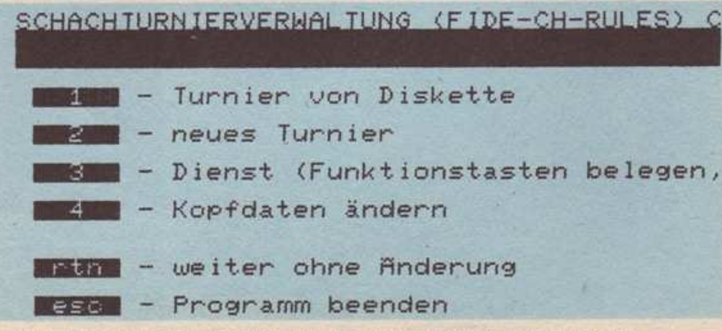


Bild 1. Das Hauptmenü der Schachverwaltung deutet die

```
6570 GOTO 60                                               <258>
6580 PRINT "(2HOME,CLR,RVSON) B I S K E T T E(2SP
     ACE)F O R M A T I E R E N (RVOFF)": POKE 229
     ,1                                                    <KNQ>
6590 PRINT "(CLR,2DOWN)BIST DU SICHER (J/N) ": GE
     T KEY A$: IF A$="J" THEN 6610                         <S2B>
6600 IF A$="N" THEN 60: ELSE 6580                          <PR8>
6610 PRINT "(2DOWN)MIT NEUER ID FORMATIEREN...JA/
     NEIN ?"                                               <DN7>
6620 GET KEY A$: IF A$<>"J" AND A$<>"N" GOTO 6610         <I7T>
6630 INPUT "(2DOWN)NEUER DISKETTENAME(2SPACE)";N$         <3JA>
6640 IF A$="J" THEN PRINT "(DOWN)WIE LAUTET NEUE
     ID ";: INPUT OD$                                      <BKO>
6650 IF A$="N" THEN OPEN 1,8,15,"N:"+N$: CLOSE 1:
      GOTO 6670                                            <CF4>
6660 OPEN 1,8,15,"N:"+N$+","+OD$: CLOSE 1                 <JF2>
6670 CLOSE 1: GOTO 60                                      <KVO>
6680 PRINT "(2HOME,CLR,RVSON,3SPACE)F(3SPACE)A(4S
     PACE)R(4SPACE)B(4SPACE)W(4SPACE)A(4SPACE)H(4
     SPACE)L(2SPACE,RVOFF)"                                <NLE>
6690 PRINT "(5DOWN,2SPACE)[ 0 ].........HINTERGRU
     ND WAEHLEN                                            <SUJ>
6700 PRINT "(DOWN,2SPACE)[ 1 ].........ZEICHENFAR
     BE WAEHLEN                                            <LBE>
6710 PRINT "(DOWN,2SPACE)[ 4 ].........RAHMENFARB
     E WAEHLEN                                             <D32>
6720 PRINT "(DOWN,2SPACE)[ * ].........M E N U E          <NS3>
6730 PRINT "(6DOWN) >>> BITTE TASTE GEDRUECKT HAL
     TEN <<<                                               <9N7>
6740 PRINT "(3DOWN)"                                       <ELK>
6750 F=1: H=0: FOR F=1 TO 16: FOR H=0 TO 7                <I6L>
6760 GET KEY A$                                            <2GD>
6770 IF A$="0" THEN B=0: GOTO 6830                        <HP1>
6780 IF A$=CHR$(49) THEN B=5: GOTO 6810                   <PLM>
6790 IF ASC(A$)=52 THEN B=4: GOTO 6830                    <JKA>
6800 IF A$="*" THEN 60: ELSE 6760                          <KAG>
6810 POKE 228,23: POKE 229,22                              <3CP>
6820 PRINT " (RVSON) ............ZEICHENFARBE....
     ..........(RVOFF)"                                    <GTJ>
6830 COLOR B,F,H: COLOR(B=0)*(-6),F                        <2VJ>
6840 NEXT H,F                                              <KSV>
6850 IF F>16 OR F<1 THEN 6750                              <AVD>
6860 GOTO 6760                                             <680>
6870 PRINT "(2HOME,CLR,RVSON) D(2SPACE)A(2SPACE)T
     (2SPACE)U(2SPACE)M(2SPACE)/(3SPACE)U(2SPACE)
     H(2SPACE)R(2SPACE)Z(2SPACE)E(2SPACE)I(2SPACE
     )T (RVOFF)"                                           <SD2>
6880 INPUT "(3DOWN)DATUM (TT.MM.JJJJ )";DA$: IF D
     A$="" OR LEN(DA$)<>10 THEN DA$="TT.MM.JJJJ"          <LNP>
6890 INPUT "(3DOWN)UHRZEIT ( HHMMSS ) ";TI$               <7JS>
6900 GOTO 60                                               <JR8>
6910 TRAP 6910: IF ER=20 THEN CHAR ,26,7,"ERROR":
      GET KEY G$: CHAR ,26,7,"(5SPACE)": A$="A":
     GOTO 4710                                             <D23>
6915 PRINT "(2HOME,CLR)": IF ER=5 THEN CHAR ,3,12
     ,"GERAET IST NICHT BETRIEBSBEREIT !!": GOTO
     6960                                                  <T8E>
6920 IF ER=-1 THEN 6940                                    <7B8>
6930 IF ER=30 THEN 6940                                    <0R8>
6940 PRINT "(2HOME,CLR)"                                   <H1Q>
6950 CHAR ,12,12,"ZURUECK INS MENUE.!!!"                   <PQB>
6960 CLOSE 1: FOR U=1 TO 1000: NEXT U: R$(W)="":
     POKE 808,112: RESUME 60                               <AGB>
6970 CLOSE 1                                               <SAN>
6980 CLOSE 1: PRINT "(RVSON,5SPACE)BITTE DRUECKEN
      SIE EINE TASTE !(4SPACE,RVOFF)": GET KEY A$
     : RETURN                                              <KID>
6990 PRINT "(CLR,4DOWN)": PRINT TAB(8)P1$: PRINT
     TAB(8)"-NACH BEENDIGUNG BITTE-"                       <481>
7000 PRINT TAB(8)P2$: PRINT TAB(8)"-(3SPACE)TASTE
      DRUECKEN !(2SPACE)-": PRINT TAB(8)P4$: RETU
     RN                                                    <51H>
7010 SU=0: S=0: SO=0: EI=0: FOR I=1 TO N: S=VAL(S
     $(I)): M$=MID$(T$(I),1,1): I$=STR$(S)                <I80>
7020 IF M$="-" THEN SU=SU-VAL(I$): ELSE SU=SU+VAL
     (I$)                                                  <4S4>
7030 IF M$="-" THEN SO=SO+VAL(I$): ELSE EI=EI+VAL
     (I$)                                                  <K45>
7040 NEXT I                                                <D94>
7050 RETURN                                                <PE7>
7060 RU=0: FOR I=1 TO O                                    <HMV>
7070 R2$=MID$(L2$(I),13,4): R1$=MID$(L2$(I),1,1)          <HQ2>
7080 IF R1$="-" THEN RU=RU-VAL(R2$): ELSE RU=RU+V
     AL(R2$)                                               <H0C>
7090 NEXT I                                                <544>
7100 RETURN                                                <907>
7110 E$="": CHAR ,26,7,"(RVOFF,10SPACE,RVSON)": R
     ETURN                                                 <VEG>
```

Listing 1. »HAUSHALT« (Schluß)

tung für Profis

erscheint das Hauptmenü (Bild 1). Sinnvoll ist es, wenn Sie nun den Menüpunkt 2 anwählen. Mit diesem ist ein neues Turnier anzulegen. Drücken Sie die Taste <2>, so erscheint eine neue Zeile, in der der Name des Turniers sowie die zu spielende Rundenzahl (jeweils gefolgt von <RETURN>) festzulegen sind. Ist dies geschehen, erscheint eine neue Eingabezeile, in der Sie den Namen eingeben, unter dem das Turnier auf Diskette gespeichert wird. Schließen Sie die Eingabe mit <RETURN> ab, speichert die Schachverwaltung Ihr Turnier.

Anschließend erscheint eine neue Bildschirmmaske. Hier nehmen Sie die Spielerverwaltung vor. Zuerst sind natürlich die Daten aller teilnehmenden Spieler einzutragen. Wählen Sie <k> (Kommando) und dann <n> (Neueingabe). Der blinkende Cursor im Eingabefeld signalisiert, daß Spieler eingegeben werden können. Drücken Sie einmal <RETURN>, so befinden Sie sich im Namensfeld. Tragen Sie nun den ersten Spieler ein. Der Nachname ist durch ein Komma mit nachfolgendem <SPACE> anzufügen. Sie können den Namen mit den Editiertasten (Cursor-Tasten usw.) wie gewohnt bearbeiten. Drücken Sie erneut <RETURN>, springt der Cursor in die zweite Rubrik, in der Sie den Verein des Spielers eintragen können. Die nächsten beiden Rubriken sind für die Eintragung des Platzes in der Rangliste, den der Spieler belegt, vorgesehen. Vereins-

128 ** (w) Reinhard A. Schulz Dtmd 1986

Hauptmenue

Diskettenverwaltung, etc.)

Leistungsvielfalt nur an

Casino-Open

\> Pairings & Results / Round 9 <

1	Hickl, Jörg	- Damljanovic, Branco		remis
2	King, Daniel John	- Suba, Michail		0:1
3	Doncev, Dimitar	- Kudrin, Sergei		remis
4	Cvitan, Ognjen	- Velikov, Petar		1:0
5	Kirov, Nino	- Klinger, Josef		0:1
6	Csom, Istvan	- Ninov, Kiril		0:1
7	Gutman, Lev	- Riemersma, Liafbern		1:0
8	Barlov, Dragan	- Mednis, Edmar		remis
9	Radulov, Ivan	- Watson, William		remis
10	Gheorghiu, Florin	- Kraut, Rainer		1:0
11	Pils, Walter	- Sokolov, Ivan		0:1
12	Cebalo, Miso	- Voiska, Magarita Boyanova	W	1:0
13	Averbach, Yuri	- Dumpor, Atif		1:0
14	Bischoff, Klaus	- Lendwai, Reinhard		remis
15	Jukic, Mirko	- Petrovic, Slavko		remis
16	Zorman, Vojko	- Blauert, Jörg		remis
17	Van Heste, Jeroen	- Farago, Ivan		0:1
18	Wegerer, Fred	- Rukavina, Josip		0:1
19	Raicevic, Vladimir	- Tabor, Josef		1:0
20	Guyot, Philippe	- Kozul, Zdenko		0:1
21	Srch, Josef	- Palosch, Osman		0:1
22	Halsegger, Herbert	- Wittmann, Dr. Walter		0:1
23	Fauland, Alex	- Kranzl, Peter		0:1
24	Rohler, Sigfried	- Witke, Thomas		remis
25	Szalanczy, Emil	- Deustch, Ludwig		0:1
26	Wiley, Tom	- Medancic, Rikard		remis
27	Fischer, D.I. Johann	- Nickl, Prof. Klaus		remis
28	Khaled, Mahdy	- Strauß, Prof. Anton		1:0
29	Reichmann, Egon	- Ebner, Hubert		remis
30	Schulze, Ulrich	- Frosch, Erich		1:0
31	Bachler, Reinhard	- Habibi, Ali		1:0

Bild 2. Spieler und Ergebnisse sind schnell eingetragen, die Erstellung der Tabellen übernimmt das Programm

spieler kennen die Bezeichnungen »INGO« und »ELO«, die die Bewertung eines Spielers in der Weltrangliste kennzeichnen. Sind Sie in der »ELO«-Spalte angelangt, erscheint in der Fußzeile des Bildschirms der Hinweis, daß mit <i> die informelle ELO-Zahl in den INGO-Wert umgerechnet wird. Die Eingabe der Wertung in der Rangliste kann allerdings auch vollständig weggelassen werden.

Geben Sie nacheinander auf die gleiche Weise alle Teilnehmer ein (Bild 3). Die Eingabe schließen Sie mit <ESC> ab.

Wieder erscheint das Untermenü mit den Punkten »Kommando«, »Turnierverwaltung«, »Funktionstasten belegen« und »abspeichern«.

Wählen Sie den Punkt »Kommando« und anschließend <l> für die Spielerliste. Mit <p> und <P> blättern Sie in der Liste seitenweise vor- und rückwärts. Mit den Tasten <+> und <-> im abgesetzten Tastenfeld blättern Sie einzeln in der Spielerliste.

Nach diesem ersten Exkurs in die Bedienung des Programms kommen wir nun zur systematischen Beschreibung der verschiedenen Programmpunkte der Schachverwaltung. Das Programm ist in weiten Teilen selbsterklärend, so daß wir die Funktionen nur kurz aufzählen. Sie können aus jedem Menü mit der Taste <ESC> zum übergeordneten Menü zurückkehren.

ECU-Championship Girls under 16

> List of Players <

Nr.	(Title) Name, Forename	Club/Town/Nation	Ingo	Elo
1	Jensen, Christine	DEN		1862
2	Moffat, Leoni	SCO		1470
3	Frauteau, Sabine	FRA		1900
4	van Amerongen, Marleen	NLD		1850
5	Paasikangas, Johanna	FIN		2005
6	Jahnke, Katrin	FRG		1520
7	Gratasdottir, Gudfridur Lilia	ISD		1520
8	Wikner, Maria	SWE		1570
9	Micklander, Elisabeth	DEN		1661
10	Bojkovic, Natasja	JUG		2220
11	Berntsen, Sheila Barth	NOR		1141
12	Smiechowska, Johanna	POL		1850

Bild 3. Die Spielerliste eines Turniers ausgedruckt auf einem NEC P 2200 in NLQ-Schrift

ECU-Championship Girls under 16

> Ranking List after the 5. Round <

Rank	Player	Points	Bhz	mBhz
1.	Bojkovic, Natasja	5		
2.	Frauteau, Sabine	3.5		
3.	Smiechowska, Johanna	3	15.5	9
4.	van Amerongen, Marleen	3	15.5	8.5
5.	Gratasdottir, Gudfridur Lilia	3	8.5	
6.	Wikner, Maria	2.5	12.5	
7.	Paasikangas, Johanna	2.5	12	
8.	Jensen, Christine	2	15	8.5
	Micklander, Elisabeth	2	15	8.5
10.	Moffat, Leoni	1.5	10.5	6.5
	Berntsen, Sheila Barth	1.5	10.5	6.5
12.	Jahnke, Katrin	.5		

Bild 4. Die Rangliste nach fünf Runden mit Buchholzwertung

Hauptmenü

<1>: Turnier von Diskette laden
<2>: neues Turnier anlegen
<3>: »Dienst«-Menü aufrufen
<4>: Kopfdaten ändern, das heißt, die Bezeichnung des Turniers in der Titelzeile neu festlegen
<rtn>: Daten speichern, weiter ohne Änderung
<esc>: Programm mit der ESC-Taste verlassen. Es erscheint eine Sicherheitsabfrage, die Sie mit <j> oder <n> beantworten können.

Alle Funktionen auf einen Blick

Dienstmenü

<1>: Funktionstasten belegen. Die Funktionstasten sind mit Begriffen, auch Steuersequenzen Ihrer Wahl zu belegen.
<2>: Inhaltsverzeichnis der Datendiskette zeigt alle gespeicherten Turniere an.
<3>: Die im Laufwerk mit der Nummer 8 eingelegte Diskette wird zweiseitig formatiert. Es sind Diskettenname und ID einzugeben.
<4>: Turnierdatei auf der Diskette im Laufwerk mit der Nummer 8 löschen. Die Turniere auf der Diskette werden angezeigt, nach dem Drücken einer Taste ist die laufende

Nummer der Turnierdatei anzugeben, es erscheint eine Sicherheitsabfrage.
<5>: Turnierdatei umbenennen, der Ablauf erfolgt wie bei Punkt 4.
<6>: sonstige Diskettenbefehle eingeben, das Befehlsformat entspricht den Angaben im Floppy-Handbuch

Mit einer anderen Taste als den Zahlen 1 bis 6 erfolgt der Sprung zum Hauptmenü.

Nachdem Sie ein Turnier von Diskette geladen haben, gelangen Sie in die eigentliche Verwaltung. Wenn Sie die Spieler eingegeben haben (s.o.), sind zuerst die Paarungen für die erste Runde einzugeben. Diese Funktion wählen Sie mit <p>: Sie haben nach dieser Wahl folgende Möglichkeiten:
<n>: Paarungsneueingabe
<a>: Paarungen aufheben
<l>: Liste der Paarungen
<f>: Spieler suchen – geben Sie den Namen des Spielers an. Anschließend wird, falls der Spieler gefunden wird, die Paarung mit diesem in der aktuellen Runde angezeigt. Sind mehrere Spieler mit diesem Namen beteiligt, werden alle betreffenden Paarungen ausgegeben.
<p>: noch nicht gepaarte Spieler anzeigen

Mit <ESC> gelangen Sie wieder in das Verwaltungsmenü.

Dort finden Sie als weiteren Punkt <e>, mit dem Sie die Ergebnisverwaltung (Bild 2) vornehmen. Folgende Punkte stehen Ihnen nach der Wahl der Ergebnisverwaltung zur Verfügung:
<n>: Ergebnisse neu eingeben. Sind in der aktuellen Runde alle Ergebnisse eingetragen, erscheint ein Hinweis in der Fußzeile des Bildschirms. Die Kennziffern aller möglichen Ergebnisse erscheinen in der Fußzeile, es ist nur die betreffende Ziffer zu drücken (<0> bis <7>).
<L>: alle Ergebnisse löschen. Die Ergebnisse der aktuellen Runde werden nach einer Sicherheitsabfrage komplett gelöscht.
<e>: offenstehende Ergebnisse der aktuellen Runde anzeigen
<l>: Liste aller offen stehenden Partien (noch nicht gespielte oder Hängepartien) ausgeben
<f>: Spieler suchen (wie oben)

Soweit zu den Funktionen im Ergebnismenü.

Das Verwaltungsmenü besitzt weitere Funktionen. Mit <n> gelangen Sie in die nächste zu spielende Runde. Der Ablauf dort ist wieder wie oben beschrieben. Mit <r> wechseln Sie in eine beliebige andere Runde. Geben Sie einfach die Rundennummer ein.

Im Untermenü »Tabellen« ist der Stand aller Spieler in der Gesamtwertung auszugeben. Sie können wählen:
<f>: Fortschrittstabelle auf Drucker
<t>: Rangtabelle ohne Buchholzwertung (siehe Kasten)
<T>: Rangtabelle mit Buchholzwertung

Bei Ausdrucken können Sie wählen, ob die Datei auf den Drucker oder in eine sequentielle Datei ausgegeben wird. Haben Sie »Drucker« gewählt, ist noch anzugeben, ob der Ausdruck in NLQ-Schrift erfolgen soll (Bild 3).

Mit der Wahl von <s> im Verwaltungsmenü gelangen Sie in die Spielerverwaltung. Sie können Spieler löschen, Spieler suchen oder die Liste aller Teilnehmer sortiert nach Namen, Verein oder Kennzahlen ausgeben. Haben Sie alle Operationen abgeschlossen, wechseln Sie mit <t> in die Turnierverwaltung.

Wie schon oben erwähnt sind die Funktionen der Schachverwaltung zum großen Teil selbsterklärend. Sie werden dies beim Umgang mit dem Programm feststellen. Wir wünschen Ihnen beim Verwalten Ihrer Schachturniere viel Erfolg. (R.A. Schulz/rs)

Buchholz-Wertung

Die Wertung nach Buchholz (Bholz) verfeinert das Schweizer System. Es werden nicht nur die bisher errungenen Punkte im Turnierverlauf bei der Ansetzung der Paarungen der nächsten Runde berücksichtigt, sondern gleichzeitig die Spielstärke der bisherigen Gegner. Zur Ermittlung der Spielstärke dienen dabei die bisher erreichten Punkte, die alle Gegner bis zur letzten Runde erreichten. Diese Punktsumme ist ein weiteres Kriterium zur Ermittlung der nächsten Gegner. Bei der Auswertung nach dem »mittleren Buchholzwert« (mBholz) fallen die Punkte des Gegners mit den meisten und die des Gegners mit den wenigsten Punkten aus der Berechnung (Bild 4).

Eingabehinweise

Die Schachverwaltung umfaßt insgesamt sechs Listings. Geben Sie bitte die Listings 1 und 2 mit dem Checksummer 128 (Seite 158) ein. Die Listings 3 bis 6 sind im 64'er-Modus mit dem MSE einzugeben und zu speichern. Kopieren Sie anschließend alle Listings auf eine Diskette, die zweiseitig formatiert ist. So steht Ihnen für die Turniere reichlich Speicherplatz zur Verfügung.

Starten Sie das Programm, indem Sie das Programm »LADER« mit

```
RUN"LADER"
```

aufrufen. Die folgenden Programmteile werden nach einem Tastendruck automatisch nachgeladen.

Beachten Sie bitte, daß der Ausdruck der Listen nur über ein Verbindungskabel am User-Port des C 128 zum Drucker erfolgt. Der angeschlossene Drucker muß also eine Centronics-Schnittstelle besitzen.

So entstand die Schachverwaltung

Als die Dortmunder Schachtage immer größer wurden, fragte sich der Schiedsrichter, wie er einen immer mehr anwachsenden Spielerberg beherrschen sollte. Da kam ein Schachfreund auf die rettende Idee: Reinhard Schulz vom ausrichtenden Club Schachfreunde Dortmund-Brackel 30. Er hatte die Vision, daß ein Turnierspielsystem wie das sogenannte »Schweizer System« für die Auswertung mit einem Computer wie geschaffen sein müßte. Reinhard Schulz besaß einen C 128 – sollte dieser Computer den Anforderungen genügen? Fast jeder Beobachter war skeptisch. Waren doch die Versuche mit einem der »großen« Computer, zum Beispiel einem IBM-PC, kläglich am Zeitproblem gescheitert. Eine Zeitersparnis von zirka 20 Minuten gegenüber der Handauslosung, die ungefähr 45 Minuten in Anspruch nahm, wäre nicht der verdiente Lohn des Programmieraufwandes gewesen.

Mit diesen Voraussetzungen ging es an die Arbeit. Drei Monate intensive Beschäftigung, Eindringen in die »Sphären« des Betriebssystems des C 128 und Einarbeiten in den nötigen Algoritmus führte zum ersten Entwurf.

Das Schweizer System

Das Schweizer System ist eine bei Schachfreunden in aller Welt beliebte Form der Ansetzung von Spielpaarungen in einem Turnier. Für einen Sieg gibt es im Schachturnier einen Punkt, für ein Remis oder Patt (Unentschieden) einen halben Punkt. Beim Schweizer System spielen immer die Spieler gegeneinander, die im bisherigen Turnierverlauf die gleiche Anzahl von Punkten errungen haben. Die erste Runde wird nach den Vorleistungen (Rangliste) der Spieler angesetzt. Natürlich muß noch berücksichtigt werden, daß möglichst nicht Gegner wiederholt gegeneinander antreten. Dieses Spielsystem führt in den meisten Fällen dazu, daß nach schon einer geringen Anzahl von Paarungen der wirkliche Leistungsstand der Spieler ermittelt wird.

Dann folgte die Umsetzung des Konzeptes in eine dem Computer verständliche Sprache. Eine Hochsprache, beispielsweise Pascal, fiel für die Umsetzung aus – viel zu langsam. Also: Aufbruch in die Welt der Maschinensprache. Assembler laden, Programm eingeben, austesten und... Absturz. Unzählige Tage und Nächte verfluchte der Autor sich und den Computer. Aber er gab nicht auf, so daß nach vielen Monaten kurz vor Weihnachten die erste lauffähige Version vorlag.

Bei einem Blitzturnier im Heimatverein sollte die Generalprobe erfolgen. Der erste Versuch scheiterte – Absturz. Doch man kam dem Fehler schnell auf die Spur: Eine Betriebssystem-Erweiterung vertrug sich nicht mit den massiven Eingriffen in die Interpreter- und Kernelroutinen. Bei den nächsten Probeläufen würdigten Vereinskameraden schon fasziniert die blitzschnelle Arbeit des C 128.

Sternstunde für den C 128

Dem großen Auftritt des C 128 stand nun nichts mehr im Weg. Die 15. Internationalen Dortmunder Schachtage nahten mit Riesenschritten. Jürgen Grastat, Vorsitzender des Schachclubs und Hauptorganisator, gab sein »O.K.«. Die folgenden Wochen wurden genutzt, um der Bedienerfreundlichkeit den letzten Schliff zu geben.

Bei Anmeldeschluß staunten die Organisatoren über die große Teilnehmerzahl. 212 Schachfreunde aus aller Welt hatten sich angemeldet, um an dem zweitgrößten Turnier in der Bundesrepublik teilzunehmen.

Wieder gab es Zweifel an der Leistungsfähigkeit des Programms auf dem C 128. Sollte die Verwaltung diese gewaltige Spielerzahl bewältigen? Es gelang. Der anwesende internationale Schiedsrichter hätte für die Festlegung der Paarungen zirka zwei Stunden benötigt. Mit Hilfe des Computers dauerte es genau zwei Sekunden! Zeitweise bildete sich eine große Zuschauertraube um den Computertisch, um die Auslosung »byte-frisch« zu erfahren.

Die Leistungsfähigkeit wurde bei dem Hauptturnier noch nicht vollständig klar. Hier wurde jeden Tag eine Partie mit insgesamt fünf Stunden Bedenkzeit gespielt. Beim folgenden Blitzturnier jedoch, bei dem jedem Spieler nur fünf Minuten Bedenkzeit pro Partie zur Verfügung stand, erhielt das Programm dann mehrfach Szenenapplaus. Das Blitzturnier wurde nach dem berühmten Schweizer System durchgeführt – ein Novum in der Geschichte eines wichtigen Turniers. Nur mit Hilfe des C 128 war die Ansetzung der Paarungen und die Auswertung der Ergebnisse in dieser Kürze zu bewältigen.

Auf dem Atari ST läuft inzwischen ebenfalls eine Version, auch die Umsetzung auf andere Computer ist geplant. Die Atari-Version vertreibt der Autor inzwischen.

Info: Reinhard Andreas Schulz, Lindemannstr. 66, 4600 Dortmund 1, Tel. 0231/136912

```
1 FAST : POKE 0,PEEK(0) OR 64: POKE 1,0: POKE 48,
  70: POKE 47,0: CLR : COLOR 6,1: GRAPHIC CLR : G
  RAPHIC 5,1: PRINT "{GREEN,CTRL+N,CTRL+K}";: TRA
  P 1001                                          <O3M>
10 BLOAD "CH.SYS1" ON B1: BLOAD "CENTRONICS" ON B
   0: BLOAD "SCREEN" ON B0: BLOAD "SPSORT" ON B0  <NQD>
100 JN$="{RVSON} J {RVOFF} / {RVSON} N {RVOFF}":
    ZA$="0123456789"                              <UHN>
102 BS$=" ABCDEFGHIJKLMNOPQRSTUVWXYZFCXVDRSWHJLYU
    ABCDEFGHIJKLMNOPQRSTUVWXYZ+=-<>!$%&/()=?[↑π+*
    ]£#'-.": CL$=CHR$(27)+"@": CR$=CHR$(27)+"Q"   <SSB>
105 DIM E$(3),EG$(9),SR%(254),NA$(254,3),NA%(254,
    1),FA$(3)                                     <OCO>
106 EG$(0)="{8SPACE}": FA$(0)=" ": FA$(1)="W": EG
    $(1)="{2SPACE}1:0{2SPACE}": EG$(2)="{2SPACE}0
    :1{2SPACE}": EG$(9)="{2SPACE}0{3SPACE}KL": FA
    $(2)="S": EG$(3)=" REMIS{2SPACE}": EG$(4)=" H
    FNGT{2SPACE}"                                 <B00>
111 EG$(5)="{2SPACE}1:0 KL": EG$(6)="{2SPACE}0:1
    KL": EG$(7)="{2SPACE}0:0 KL": EG$(8)="{2SPACE
    }1{3SPACE}KL"                                 <7H2>
116 TA%=15156: LT%=15974: LE%=15986: UP%=16811: G
    T%=16018                                      <UM6>
120 XE$=" {RVSON} 0 {RVOFF} - NOCH NICHT GESPIELT
    {2SPACE}*{2SPACE,RVSON} 1 {RVOFF} - 1:0{2SPAC
    E}*{2SPACE,RVSON} 2 {RVOFF} - 0:1{2SPACE}*{2S
    PACE,RVSON} 3 {RVOFF} - REMIS "               <RSN>
121 YE$=" {RVSON} 4 {RVOFF} - HFNGT{2SPACE}*{2SPA
    CE,RVSON} 5 {RVOFF} - 1:0 KL{2SPACE}*{2SPACE,
    RVSON} 6 {RVOFF,2SPACE}0:1 KL{2SPACE}*{2SPACE
    ,RVSON} 7 {RVOFF,2SPACE}- 0:0 KL{2HOME}"      <LL1>
122 DF$="FORTSCHRITTSTABELLE WIRD GEDRUCKT! - DRU
    CKSTOPP MIT {RVSON} ESC {RVOFF}"              <ASV>
124 SP$="": FOR II=1 TO 80: SP$=SP$+" ": NEXT II  <NVF>
125 BX$=" {RVSON} + {RVOFF} / {RVSON} - {RVOFF} -
    EINZELN ... *{2SPACE,RVSON} P {RVOFF} / {RVS
    ON} P {RVOFF} - SEITENWEISE BLFTTERN": DL$="
    SOLLEN DATEN GESICHERT WERDEN?{SHF.SPACE}"    <2AA>
214 GOSUB 10001: GOTO 63901                       <2S9>
1001 IF ER=30 THEN RESUME                         <2H1>
1002 IF ER=2 THEN CLOSE 2: CLOSE 15: RESUME       <HI2>
1003 BANK 15: SYS 5760: GOSUB 2006: PRINT "{2HOME
     }": CHAR ,0,23,CL$: PRINT " FEHLER - ";ERR$(
     ER)                                          <DID>
1004 PRINT " {RVSON} RTN {RVOFF} - FORTSETZUNG DE
     S PROGRAMMS{2SPACE}*{2SPACE,RVSON} ESC {RVOF
     F} - NEUSTART";                              <0UU>
1005 GET KEY XT$: IF XT$=CHR$(13) THEN 1014       <APH>
1006 IF XT$<>CHR$(27) THEN 1005                   <V48>
1007 CHAR ,0,23,CL$                               <HQ2>
1008 PRINT " SIND SIE SICHER? "JN$" VORSICHT - DA
     TENVERLUST!"                                 <JB9>
1009 GET KEY XT$: XT%=INSTR("JN",XT$): ON XT% GOT
     O 1010,1004: GOTO 1009                       <7AJ>
1010 RUN 1011                                     <9ND>
1011 BANK 15: SYS 5768: POKE 48,70: POKE 47,0: CL
     R                                            <53K>
1012 TRAP 1001                                    <IAS>
1013 GOTO 100                                     <JS6>
1014 BANK 15: SYS 5768: RESUME                    <6BM>
2001 VOL 15: SOUND 1.6000.10....2: RETURN         <GR9>
2006 VOL 15: SOUND 1,1500,20,,,,1: RETURN         <EP1>
3001 PRINT "{2HOME}": WINDOW EX%,EY%,EX%+EL%,EY%,
     1: PRINT CHR$(27)"M";: BANK 15: SYS 52591    <J60>
3005 PRINT CHR$(27)"J"E$(0);CHR$(27)"Q";          <EM6>
3010 GET KEY XT$: IF XT$="{CTRL+I}" THEN 3005     <38A>
3015 IF INSTR(E$(2)+CHR$(27),XT$)>0 THEN 3060     <FHP>
3020 IF XT$=CHR$(13) THEN 3050                    <TRE>
3025 IF XT$=CHR$(20) AND POS(0)=0 THEN PRINT " ";
     : GOTO 3045                                  <EB4>
3030 IF INSTR("{LEFT,RIGHT,CLR}"+CHR$(20),XT$)>0
     THEN 3045                                    <FLB>
3031 IF XT$="{INST}" THEN PRINT "{INST} {LEFT}";:
      GOTO 3010                                   <NOP>
3032 IF XT$="{HOME}" THEN PRINT CHR$(27)"J";: GOT
     O 3010                                       <NUB>
3035 IF POS(0)=EL% THEN 3010                      <2LH>
3040 IF INSTR(E$(3),XT$)=0 THEN GOSUB 2006: GOTO
     3010                                         <ABT>
3045 PRINT XT$;: GOTO 3010                        <K7J>
3050 SYS 52639: PRINT CHR$(27)"K{LEFT}";          <SQ1>
3055 E$(1)="": OPEN 3,3: FOR II=POS(0) TO .STEP -
     1: GET #3,XT$: E$(1)=XT$+E$(1): PRINT "{2LEF
     T}";: NEXT II: CLOSE 3                       <3R6>
3056 IF E$(1)=" " THEN E$(1)=""                   <IT1>
3060 SYS 52639: PRINT CHR$(27)"L{2HOME}": RETURN  <B8I>
4001 CHAR ,0,23,CL$: PP$=DS$: IF DS=0 THEN PP$="6
     3, FILE EXISTS,00,00"                        <475>
4005 GOSUB 2006: PRINT " DISKETTENFEHLER - "PP$:
     CLOSE 2: CLOSE 15                            <TOO>
4010 PRINT " {RVSON} ESC {RVOFF} - ZURXCK *** {RV
     SON} RTN {RVOFF} - WEITER";                  <6RR>
4020 GET KEY XT$: IF INSTR(CHR$(27)+CHR$(13),XT$)
     =0 THEN 4020                                 <2T3>
4025 CHAR ,0,23,CL$: IF XT$=CHR$(27) THEN CR%=1:
     RETURN                                       <M0R>
4030 CR%=0: RETURN                                <OAF>
4101 OPEN 15,8,15,"I0"                            <NPE>
4102 IF DS=0 THEN 4104                            <S0P>
4103 GOSUB 4001: IF CR%=1 THEN RETURN : ELSE GOTO
      4101                                        <6VU>
4104 OPEN 2,8,2,"#"                               <R06>
4105 PRINT#15,"U1:2 0 18 0"                       <H6C>
4106 PRINT#15,"B-P";2;144                         <AQ9>
4107 YD$="": FOR II=144 TO 163: GET #2,XD$: YD$=Y
     D$+XD$: NEXT II                              <O7H>
4108 XD$=RIGHT$(YD$,2): YD$=LEFT$(YD$,INSTR(YD$,C
     HR$(160))-1)                                 <ANP>
4109 IF DS=0 THEN CLOSE 2: CLOSE 15: CR%=0: RETUR
     N                                            <UBL>
4110 GOSUB 4001: IF CR%=1 THEN RETURN : ELSE GOTO
      4101                                        <6RU>
4121 GOSUB 4101: IF CR%=1 THEN RETURN             <IBA>
4122 ID$=XD$: DN$=YD$: RETURN                     <PHF>
4131 GOSUB 4101: IF CR%=1 THEN RETURN             <GBA>
4132 IF YD$=DN$ AND XD$=ID$ THEN CR%=0: RETURN    <OH1>
4133 CHAR ,0,23,CL$: GOSUB 2006: PRINT " FALSCHE
```

Listing 1. »CH-VERWALTUNG« ist das Hauptprogramm. Dieses lädt die Maschinenroutinen (Listings 3 bis 6) nach. Bitte mit dem Checksummer 128 eingeben.

```
     DISKETTE!": GOSUB 4010: IF CR%=0 THEN 4131       <SJV>
4234 RETURN                                           <PM7>
5001 CR%=0: BANK 15: SYS DEC("0D7D"): IF PEEK(DEC
     ("0DBA"))=0 THEN RETURN                          <FVS>
5002 CR%=1: GOSUB 2006: CHAR ,0,23,CL$: PRINT " D
     RUCKER EINSCHALTEN - PAPIER EINLEGEN!": GET
     KEY XT$: IF XT$=CHR$(27) THEN RETURN : ELSE
     GOTO 5001                                        <24P>
5011 PM%=0: CR%=0: CHAR ,0,23,CL$: PRINT " AUSDRU
     CK IN NLQ-SCHRIFT? "JN$: GET KEY XT$: IF XT$
     ="J" THEN PM%=16                                 <ABL>
5012 IF XT$=CHR$(27) THEN CR%=1                       <M30>
5013 RETURN                                           <1L7>
5021 CR%=0: CHAR ,0,23,CL$: PRINT " (RVSON) D (RV
     OFF) - AUSGABE AUF DRUCKER(2SPACE)*(2SPACE.R
     VSON) S (RVOFF) ... IN SEQUENTIELLE DATEI"       <R74>
5022 GET KEY XT$: XT%=INSTR("DS"+CHR$(27),XT$): I
     F XT%=0 THEN 5022                                <237>
5023 RETURN                                           <HJ7>
5031 CR%=0: CHAR ,0,23,CL$: PRINT " (RVSON) 0 (RV
     OFF) - EINFACHE ...(2SPACE)*(2SPACE,RVSON) 1
      (RVOFF) - DOPPELTE AUSGABE"                     <GNS>
5032 GET KEY XT$: XT%=INSTR("01"+CHR$(27),XT$): O
     N XT% GOTO 5034,5034,5033: GOTO 5032             <HD7>
5033 CR%=1                                            <VRF>
5034 RETURN                                           <PM7>
10001 PRINT "(2HOME,CLR)SCHACHTURNIERVERWALTUNG(S
      HF.SPACE)(FIDE-CH-RULES)(SHF.SPACE)C128 **
      (W) REINHARD R. SCHULZ DTMD 1986": RETURN       <LA6>
10005 PRINT CHR$(27)"M": WINDOW 0,1,79,1: PRINT "
      (RVSON)"SP$                                     <273>
10006 WINDOW 0,2,79,2,1: PRINT "(RVSON)"SP$: PRIN
      T CHR$(27)"L(2HOME)": PRINT "(RVSON)" LEFT$
      (SP$,79)                                        <PFM>
10010 CHAR 1,1,1,TU$,1: CHAR ,1,2: IF DR%>0 THEN
      PRINT "(RVSON)RUNDE: "+RIGHT$(STR$(DR%),2)"
       / " RIGHT$(STR$(RD%),2)" / " RIGHT$(STR$(M
      R%),2);                                         <0J6>
10011 CHAR ,0,2: PRINT TAB(20)"(RVSON)" LEFT$(SP$
      ,59-LEN(M0$+M1$))"(RVOFF,CTRL+B)"M0$+M1$        <OBS>
10012 RETURN                                          <GC7>
17001 CHAR ,0,23,CL$: PRINT " DATEINAME FÜR DRUCK
      DATEI:"                                         <0CV>
17005 EX%=27: EY%=23: EL%=16: E$(0)="": E$(3)=ZA$
      +".ABCDEFGHIJKLMNOPQRTSUVWXYZ": E$(2)=""        <KTB>
17010 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN         <2PM>
17015 DOPEN #2,(E$(1)),W: IF DS=0 THEN 17025          <477>
17020 GOSUB 4001: IF CR%=1 THEN 17001: ELSE GOTO
      17015                                           <CU5>
17025 CR%=0: CS%=1: II=1: P1=0: H1=0: B1=0: A1=0:
       BANK 1                                         <NE0>
17030 DO WHILE II<=MS%                                <EFV>
17035 GOSUB 37700: GOSUB 37800: GOSUB 18900           <4CF>
17040 PRINT#2,XX$                                     <RJ2>
17045 II=II+1: LOOP                                   <ALP>
17050 IF DS=0 THEN DCLOSE #2: RETURN                  <6S3>
17055 GOSUB 4001: GOTO 17001                          <Q07>
18001 CHAR ,0,23,CL$: PRINT " DATEINAME FÜR DRUCK
      DATEI:"                                         <7S1>
18005 EX%=27: EY%=23: EL%=16: E$(0)="": E$(3)=ZA$
      +".ABCDEFGHIJKLMNOPQRTSUVWXYZ": E$(2)=""        <N54>
18010 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN         <UU6>
18015 DOPEN #2,(E$(1)),W: IF DS=0 THEN 18025          <L53>
18020 GOSUB 4001: IF CR%=1 THEN 18001: ELSE GOTO
      18015                                           <F7M>
18025 PN%=1: CR%=0                                    <1VM>
18030 DO WHILE PN%<=PA%                               <OR9>
18035 GOSUB 20101: GOSUB 36800                        <ILM>
18040 XX$=LEFT$(XX$+SP$,35)+" ": XX$=LEFT$(XX$+XY
      $+SP$,68)+" "+XZ$: GOSUB 18900                  <RVE>
18041 PRINT#2,XX$                                     <2V3>
18045 PN%=PN%+1: LOOP                                 <BB7>
18050 IF DS=0 THEN DCLOSE #2: RETURN                  <OS3>
18055 GOSUB 4001: GOTO 18001                          <DER>
18900 XY$="FCX±=-VRHDSJLYWU": XZ$="(CTRL+A,CTRL+B
      ,CTRL+C,CTRL+D,WHITE,CTRL+F,CTRL+G)AAEEEIOU
      U"                                              <I46>
18901 FOR IX=1 TO LEN(XY$): XT$=MID$(XY$,IX,1)        <C87>
18902 DO                                              <APF>
18903 XT%=INSTR(XX$,XT$): IF XT%=0 THEN EXIT          <5SA>
18904 MID$(XX$,XT%,1)=MID$(XZ$,IX,1)                  <6NQ>
18905 LOOP                                            <KU1>
18906 NEXT IX                                         <RH5>
18907 XY$="(CTRL+A,CTRL+B,CTRL+C,CTRL+D,WHITE,CTR
      L+F,CTRL+G)": XZ$=CHR$(165)+CHR$(182)+CHR$(
      184)+CHR$(185)+CHR$(186)+CHR$(187)+CHR$(188
      )                                               <4ME>
18908 FOR IX=1 TO LEN(XY$): XT$=MID$(XY$,IX,1)        <UAF>
18909 DO                                              <F9P>
18910 XT%=INSTR(XX$,XT$): IF XT%=0 THEN EXIT          <5C8>
18911 MID$(XX$,XT%,1)=MID$(XZ$,IX,1)                  <UJQ>
18912 LOOP                                            <UV9>
18913 NEXT IX: RETURN                                 <0F2>
19001 GOSUB 5001: IF CR%=1 THEN RETURN                <F7V>
19004 CHAR ,0,23,CL$: PRINT DF$                       <B66>
19005 OPEN 5,5: OPEN 4,4                              <NTI>
19006 PRINT#5,CHR$(27);CHR$(108);CHR$(3);             <8BO>
19007 PM%=16: PP$="> TABLE OF PLAYED ROUNDS <": G
      OSUB 35900                                      <V7B>
19010 II=1: BANK 1                                    <RSV>
19011 POKE DEC("3300"),MS%: SYS GT%                   <BSV>
19012 PRINT#5,CHR$(27)"0" CHR$(27)"!" CHR$(20);CH
      R$(27)"U1";CHR$(27)":" CHR$(0);CHR$(0);CHR$
      (0);                                            <7RV>
19013 PRINT#5,CHR$(27)"&" CHR$(0);CHR$(252);CHR$(
      252);CHR$(139);CHR$(255);CHR$(0);CHR$(129);
      CHR$(0);CHR$(129);CHR$(0);CHR$(129);CHR$(0)
      ;CHR$(129);CHR$(0);CHR$(255);                   <27N>
19014 PRINT#5,CHR$(27)"&" CHR$(0);CHR$(254);CHR$(
      254);CHR$(139);CHR$(255);CHR$(0);CHR$(255);
      CHR$(0);CHR$(255);CHR$(0);CHR$(255);CHR$(0)
      ;CHR$(255);CHR$(0);CHR$(255);                   <NA4>
19015 GOSUB 19100: XA$=CHR$(27)+"%(CTRL+A)"+CHR$(
      0)+XA$                                          <NP8>
19020 DO WHILE II<=MS%                                <TVH>
19021 GET XT$: IF XT$<>CHR$(27) THEN 19040            <THG>
19022 GOSUB 2001: CHAR ,0,23,CL$: PRINT " DRUCK I
      ST ANGEHALTEN! - SOLL DRUCK FORTGESETZT WER
      DEN? "JN$                                       <TJN>
19023 GET KEY XT$: IF XT$="N" THEN EXIT               <4U6>
19024 CHAR ,0,23,CL$                                  <0JN>
19040 XX$="(5SPACE)"+LEFT$(NA$(II,0)+SP$,14)+" ("
      +LEFT$(NA$(II,3)+SP$,3)+")"                     <QSL>
19041 XY$=RIGHT$("(2SPACE)"+STR$(II),3)+"(2SPACE)
      "+NA$(II,1): XY$=LEFT$(XY$+SP$,25)              <8IA>
19042 XZ$="(5SPACE)"+LEFT$(NA$(II,2)+SP$,20)+CHR$
      (27)                                            <BRJ>
19051 XB$=CHR$(27)+"R(CTRL+B)"+CHR$(27)+"%(CTRL+A
      )"+CHR$(0): XC$=XB$: XD$=XB$                    <A18>
19052 P0=0: FOR IK=1 TO DR%: SP%=PEEK(4096+IK+II*
      16): XX%=PEEK(8192+IK+II*16)                    <SMC>
19053 XB$=XB$+" "+CHR$(251): XC$=XC$+" "+CHR$(251
      ): XD$=XD$+" "+CHR$(251)                        <290>
19054 IF(XX% AND 192)=0 THEN XB$=XB$+"(2SPACE)"       <RM1>
19055 IF(XX% AND 192)=128 THEN XB$=XB$+CHR$(252)+
      " "                                             <5DM>
19056 IF(XX% AND 192)=192 THEN XB$=XB$+CHR$(254)+
      " "                                             <RUM>
19057 IF SP%=0 THEN XB$=XB$+CHR$(122)+CHR$(122)+C
      HR$(103): GOTO 19060                            <NGC>
19058 IF SP%=255 THEN XB$=XB$+CHR$(115)+CHR$(112)
      +CHR$(102): GOTO 19060                          <6RV>
19059 XB$=XB$+RIGHT$("(2SPACE)"+STR$(SP%),3)          <QOL>
19060 P1=(XX% AND 3)/2: XC$=XC$+RIGHT$("(2SPACE)"
      +STR$(P1),3): P0=P0+P1                          <TBB>
19062 IF(XX% AND 8)=8 THEN XC$=XC$+CHR$(107)+CHR$
      (108): ELSE XC$=XC$+"(2SPACE)"                  <AHM>
19063 XD$=XD$+RIGHT$("(5SPACE)"+STR$(P0),5)           <72R>
19064 NEXT IK                                         <8VC>
19065 PRINT#5,XA$: PRINT#4,XX$;: PRINT#5,XB$: PRI
      NT#5,"(CTRL+J)";XA$: PRINT#4,XY$;: PRINT#5,
      XC$: PRINT#5,"(CTRL+J)"XA$: PRINT#4,XZ$;: P
      RINT#5,XD$;                                     <KS5>
19066 IF INT(II/22)=II/22 THEN PRINT#4: PRINT#5,X
      A$"(CTRL+L)" CHR$(27)"2";: GOSUB 35900: PRI
      NT#5,CHR$(27)"!" CHR$(20);CHR$(27)"0";: GOT
      O 19071                                         <GRO>
19067 IF II=MS% THEN PRINT#4: PRINT#5,XA$;: EXIT      <PIN>
19068 PRINT#5,CHR$(27)"J" CHR$(2);CHR$(27)"-(CTRL
      +A)"                                            <IKS>
19069 PRINT#4,LEFT$(SP$,26);: PRINT#4,LEFT$(SP$+S
      P$,7*DR%+1)                                     <R0R>
19070 PRINT#5,CHR$(27)"J(CTRL+L)" CHR$(27)"-" CHR
      $(0)                                            <JD6>
19071 II=II+1: LOOP                                   <E9Q>
19072 PRINT#5,"(CTRL+L)";: CLOSE 4: CLOSE 5: RETU
      RN                                              <8TL>
19100 PRINT#5,CHR$(27)"&" CHR$(0);CHR$(251);CHR$(
      251);CHR$(139);CHR$(0);CHR$(0);CHR$(0);CHR$
      (0);CHR$(255);CHR$(0);CHR$(0);CHR$(0);CHR$(
      0);CHR$(0);CHR$(0);                             <6I4>
19102 XA$=CHR$(27)+CHR$(106)+CHR$(20)+LEFT$(SP$,2
      4)                                              <NDT>
19103 FOR IK=1 TO DR%: XA$=XA$+"(2SPACE)"+CHR$(25
      1)+"(4SPACE)": NEXT IK                          <GS8>
19104 XA$=XA$+CHR$(27)+"J"+CHR$(20)                   <H2I>
19105 RETURN                                          <Q57>
20001 CX%=0: BANK 1: DR%=PEEK(DEC("1000"))            <JNJ>
20002 RD%=1: DO                                       <R7A>
20003 IF PEEK(DEC("1001")+RD%)=0 THEN EXIT            <8GC>
20004 RD%=RD%+1: LOOP : IF TU%=1 THEN TU%=2: GOSU
      B 51001: IF CR%=1 THEN 63901                    <S8E>
20005 FOR IJ=1 TO MS%: XZ%=NA%(IJ,0): IF XZ%=0 TH
      EN XZ%=1000                                     <2H6>
20006 XX%=INT(XZ%/256): XY%=XZ%-XX%*256: POKE DEC
      ("0400")+IJ,XY%: POKE DEC("0500")+IJ,XX%        <HF1>
20007 XZ%=NA%(IJ,1): IF XZ%=0 AND NA%(IJ,0)<130 T
      HEN XZ%=SGN(NA%(IJ,0))*(2839-8*NA%(IJ,0))       <1HL>
20008 XX%=INT(XZ%/256): XY%=XZ%-XX%*256: POKE DEC
      ("0A00")+IJ,XY%: POKE DEC("0B00")+IJ,XX%        <COB>
20009 NEXT IJ                                         <32C>
20010 POKE DEC("04FF"),255: POKE DEC("05FF"),255:
       POKE DEC("0AFF"),0: POKE DEC("0BFF"),0: PO
      KE DEC("06FF"),0: POKE DEC("07FF"),0            <5QD>
20011 IF TU%=2 THEN POKE DEC("3300"),MS%: SYS DEC
      ("42A0")                                        <SM0>
20012 BANK 1: PA%=PEEK(DEC("3000")): GP%=PEEK(DEC
      ("3100")): SE%=PEEK(DEC("3200")): MO%=PEEK(
      DEC("1000")+DR%)                                <PLC>
20014 MM%=5: M0$="TURNIERVERWALTUNG": M1$="": GOS
      UB 10005                                        <0JU>
20015 CHAR ,0,3,CL$                                   <OH0>
20020 PRINT "(RVSON) NR. SPIELER 1(23SPACE) SPIEL
      ER 2(23SPACE) ERGEBNIS "                        <83B>
20025 CHAR 1,0,22,"T+++T+++T++++++++++++++++++++++
      +++++++T+++T++++++++++++++++++++++++++++T++
      ++++++"                                         <I0J>
20030 IF PA%>0 THEN GOSUB 20201                       <GOG>
20035 GOTO 21001                                      <8LE>
20101 BANK 1: SP%=PEEK(DEC("3000")+PN%)               <UUA>
20102 SO%=PEEK(DEC("3100")+PN%)                       <7SA>
20103 EG%=PEEK(DEC("3200")+PN%): RETURN               <348>
20111 BANK 1: POKE DEC("3000")+PN%,SP%: GP%=GP%+1     <LEV>
20112 POKE DEC("3100")+PN%,SO%: IF SO%<>255 AND S
      O%>0 THEN GP%=GP%+1                             <HUS>
20113 PA%=PA%+1: POKE DEC("3000"),PA%                 <F3E>
20114 POKE DEC("3100"),GP%                            <JS9>
20115 POKE DEC("3200")+PN%,EG%: IF EG%>0 THEN SE%
      =SE%+1                                          <75B>
20116 POKE DEC("3200"),SE%: RETURN                    <UR4>
20121 IF PN%>0 THEN PRINT RIGHT$("(4SPACE)"+STR$(
      PN%),4);                                        <29I>
20122 PRINT TAB(5) RIGHT$("(2SPACE)"+STR$(SP%),3)
      ;                                               <CA6>
20123 XB$=NA$(SP%,1): IF NA$(SP%,2)<>"" THEN XB$=
      XB$+", "+NA$(SP%,2)                             <JQI>
20124 PRINT TAB(9) LEFT$(XB$,28);                     <586>
20125 IF SO%=0 THEN XB$="ZURÜCKGEZOGEN": GOTO 201
      31                                              <CB6>
20126 IF SO%=255 THEN XB$="SPIELFREI": GOTO 20131     <4VG>
20128 IF SO%=256 THEN XB$="NOCH NICHT GEPAART": G
      OTO 20131                                       <K39>
20129 PRINT TAB(38) RIGHT$("(2SPACE)"+STR$(SO%),3
      );                                              <OCJ>
20130 XB$=NA$(SO%,1): IF NA$(SO%,2)<>"" THEN XB$=
      XB$+", "+NA$(SO%,2)                             <HIP>
20131 PRINT TAB(42) LEFT$(XB$,28);                    <OPK>
20133 PRINT TAB(71)EG$(EG%);                          <V9S>
20135 BANK 15: SYS 2848: RETURN                       <FB0>
20201 M1$=" / PAARUNGS-/ERGEBNISLISTE": GOSUB 100
      11: SR%(0)=PA%: FOR IJ=1 TO PA%: SR%(IJ)=IJ
      : NEXT IJ: IJ=1: IF CX%=1 AND PN%-7>0 THEN
      IJ=PN%-7                                        <KTE>
20202 WINDOW 0,5,79,20,1: IK=IJ                       <V9T>
```

```
20203 DO WHILE IJ<=SR%(0)                                <L37>
20204 PN%=SR%(IJ): GOSUB 20101                           <NPK>
20205 GOSUB 20121: PRINT : IF IJ=IK+14 THEN EXIT         <VGH>
20206 IJ=IJ+1                                            <QBB>
20207 LOOP : IJ=IK+14                                    <3F9>
20208 MM%=PEEK(235): PRINT "(2HOME)": IF SR%(0)=0
      THEN RETURN                                        <GAL>
20209 CHAR ,0,23,CL$                                     <5FG>
20210 PRINT " (RVSON) D (RVOFF) - LISTE AUF DRUCK
      ER": PRINT BX$;                                    <OR2>
20220 PRINT "(2HOME)": CHAR 1,0,21,CR$: GET KEY X
      T$                                                 <HH5>
20225 IF INSTR("D+-PP(HOME,CLR)",XT$)=0 THEN RETU
      RN                                                 <K53>
20230 IF XT$="D" THEN GOSUB 36001: GOTO 20209            <M73>
20235 IF XT$="+" AND IJ<SR%(0) THEN IJ=IJ+1: IK=I
      K+1: PN%=SR%(IJ): GOSUB 20101: WINDOW 0,5,7
      9,20: CHAR ,0,15,CR$: GOSUB 20121: PRINT :
      GOTO 20220                                         <0EC>
20240 IF XT$="-" AND IK>1 THEN IJ=IJ-1: IK=IK-1:
      PN%=SR%(IK): GOSUB 20101: WINDOW 0,5,79,20:
      CHAR ,0,0,CHR$(27)+"I": GOSUB 20121: GOTO
      20220                                              <2A4>
20241 IF XT$="(HOME)" THEN IJ=1: GOTO 20202              <VS3>
20242 IF XT$="(CLR)" AND SR%(0)>14 THEN IJ=SR%(0)
      -14: GOTO 20202                                    <2RB>
20243 IF XT$="P" AND IJ<SR%(0) THEN IJ=IJ+1: GOTO
      20202                                              <DML>
20244 IF XT$="P" AND IK-15>0 THEN IJ=IK-15: GOTO
      20202                                              <027>
20245 GOTO 20220                                         <9HC>
20260 M1$=" / SPIELER SUCHEN": GOSUB 10011: WINDO
      W 0,5,79,20,1: PRINT "(2HOME)": MM%=5: BANK
      1                                                  <AGP>
20265 CHAR ,0,23,CL$: PRINT " GEBEN SIE DEN NAMEN
      DES GESUCHTEN SPIELERS EIN!"                       <LFR>
20270 CR%=0: GOSUB 20370                                 <BAR>
20275 IF CR%=1 THEN RETURN                               <7SD>
20280 PN%=1: IJ=1                                        <6PH>
20285 DO WHILE PN%<=PA%                                  <EFJ>
20290 GOSUB 20101                                        <SAU>
20295 XX$=LEFT$(NA$(SP%,1)+", "+NA$(SP%,2),LEN(E$
      (1)))                                              <CJ3>
20300 XY$="+": IF SO%>0 AND SO%<255 THEN XY$=LEFT
      $(NA$(SO%,1)+", "+NA$(SO%,2),LEN(E$(1)))           <AAB>
20305 IF XX$=E$(1) OR XY$=E$(1) THEN SR%(IJ)=PN%:
      IJ=IJ+1                                            <QNK>
20310 PN%=PN%+1: LOOP                                    <JR9>
20315 SR%(0)=IJ-1: IJ=1: GOTO 20202                      <HSM>
20370 EY%=21: EX%=9: EL%=28: E$(0)="": E$(2)="":
      E$(3)=BS$+ZA$+","                                  <RF3>
20375 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: ELS
      E CR%=0                                            <MJK>
20380 RETURN                                             <H07>
20395 EY%=21: EX%=42: EL%=28: E$(0)="": E$(2)="":
      E$(3)=BS$+ZA$+","                                  <BP5>
20400 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: ELS
      E CR%=0                                            <1JK>
20405 RETURN                                             <P97>
20420 M1$=" /(2SPACE)NOCH NICHT GEPAARTE SPIELER"
      : GOSUB 10011                                      <6IV>
20425 CHAR ,0,23,CL$: WINDOW 0,5,79,20,1: PRINT "
      (2HOME)": MM%=5                                    <BML>
20430 SP%=1: SO%=256: EG%=0: IJ=1                        <DBK>
20435 DO WHILE SP%<=MS%                                  <4A0>
20440 BANK 1: POKE DEC("3820"),SP%: SYS UP%: PN%=
      PEEK(DEC("3820")): IF PN%>0 THEN 20442             <6RT>
20441 SR%(IJ)=SP%: IJ=IJ+1                               <V1F>
20442 SP%=SP%+1: LOOP : IF IJ=1 THEN RETURN              <F6G>
20443 CHAR ,0,23,CL$: PRINT " (RVSON) D (RVOFF) -
      LISTE AUF DRUCKER": PRINT BX$;: SR%(0)=IJ-
      1: IJ=1                                            <AGE>
20444 WINDOW 0,5,79,20,1: IK=IJ                          <U5T>
20445 DO WHILE IJ<=SR%(0)                                <GJH>
20446 SP%=SR%(IJ): GOSUB 20122: PRINT                    <2A4>
20447 IF IJ=IK+14 THEN EXIT                              <7IB>
20448 IJ=IJ+1: LOOP : IJ=IK+14                           <HHJ>
20449 MM%=PEEK(235): PRINT "(2HOME)": IF SR%(0)=0
      THEN RETURN                                        <IEL>
20450 PRINT "(2HOME)": GET KEY XT$: XT%=INSTR("D+
      -(HOME,CLR)PP",XT$): IF XT%=0 THEN RETURN          <I5C>
20451 ON XT% GOTO 20452,20453,20454,20456,20457,2
      0459,20461                                         <1U6>
20452 GOSUB 20503: GOTO 20449                            <L0F>
20453 IF IJ<SR%(0) THEN IJ=IJ+1: IK=IK+1: SP%=SR%
      (IJ): WINDOW 0,5,79,20: CHAR ,0,15,CR$: GOS
      UB 20122: PRINT : GOTO 20450: ELSE GOTO 204
      50                                                 <62B>
20454 IF IK>1 THEN IJ=IJ-1: IK=IK-1: SP%=SR%(IK):
      WINDOW 0,5,79,20: CHAR ,0,0,CHR$(27)+"I":
      GOSUB 20122                                        <A7C>
20455 GOTO 20450                                         <DP7>
20456 IJ=1: GOTO 20444                                   <DN5>
20457 IF SR%(0)>14 THEN IJ=SR%(0)-14: GOTO 20444         <G4I>
20458 GOTO 20450                                         <EH7>
20459 IF SR%(0)>IJ THEN IJ=IJ+1: GOTO 20444              <CIP>
20460 GOTO 20450                                         <D17>
20461 IF IK>15 THEN IJ=IK-15: GOTO 20444                 <MB0>
20462 GOTO 20450                                         <DH7>
20503 GOSUB 5001: IF CR%=1 THEN RETURN                   <1BU>
20506 PM%=0: OPEN 5,5: OPEN 4,4                          <M2K>
20507 PRINT#5,CHR$(27);CHR$(108);CHR$(7);                <764>
20508 PP$="> NOCH NICHT GEPAARTE SPIELER / RUNDE"
      +STR$(DR%)+" <"                                    <3J5>
20509 GOSUB 35900: GOSUB 36900: IX=1                     <A7I>
20511 DO WHILE IX<=SR%(0)                                <GBR>
20512 SP%=SR%(IX)                                        <AQU>
20513 XX$=RIGHT$("  "+STR$(SP%),3)+"(2SPACE)"+NA$(
      SP%,1): IF NA$(SP%,2)<>"" THEN XX$=XX$+", "
      +NA$(SP%,2)                                        <PR4>
20514 XX$=LEFT$(XX$+SP$,40)                              <BDI>
20515 PRINT#4,XX$;: IF IX/2=INT(IX/2) THEN PRINT#
      4: PP%=PP%+1                                       <JIF>
20516 IF PP%=60 THEN PRINT#5,"(CTRL+L)";: GOSUB 3
      5900: GOSUB 36900                                  <NR9>
20517 IX=IX+1: LOOP                                      <R3P>
20518 PRINT#5,"(CTRL+L)";: CLOSE 4: CLOSE 5: RETU
      RN                                                 <IEV>
20601 M1$=" / PAARUNGSNEUEINGABE": GOSUB 10011           <3H7>
20602 CHAR ,0,23,CL$: PRINT " (RVSON) H (RVOFF) -
      PAARUNGSEINGABE VON HAND(2SPACE)*(2SPACE,R
      VSON) C (RVOFF) CH-SYSTEM"                         <V5S>
20604 CHAR 1,0,21,CR$: GET KEY XT$                       <JLQ>
20605 IF XT$=CHR$(27) THEN RETURN                        <GPO>
20606 IF XT$="H" THEN 20616                              <N3V>
20607 IF XT$<>"C" OR DR%<>RD% OR GP%>=MS%-2 THEN
      GOSUB 2006: GOTO 20604                             <B0J>
20608 IJ=1: BANK 1: DO WHILE IJ<DR%                      <5CV>
20609 IF PEEK(DEC("1000")+IJ)<3 THEN GOSUB 2006:
      EXIT                                               <93A>
20610 IJ=IJ+1: LOOP                                      <57P>
20611 IF IJ<RD% THEN 20604                               <TU5>
20612 CHAR ,0,23,CL$: PRINT " PAARUNGEN WERDEN ER
      STELLT!": BANK 1: POKE DEC("3000"),PA%: POK
      E DEC("3100"),GP%: POKE DEC("3200"),SE%: PO
      KE DEC("3820"),MS%-GP%                             <S4D>
20613 POKE DEC("3300"),MS%: IF PA%>0 THEN SYS DEC
      ("4420"): ELSE SYS DEC("382E")                     <QHL>
20614 IF PEEK(DEC("3820"))=0 THEN PA%=PEEK(DEC("3
      000")): GP%=PEEK(DEC("3100")): SE%=PEEK(DEC
      ("3200")): GOSUB 2001: GOTO 20201                  <Q8K>
20615 CHAR ,0,23,CL$: GOSUB 2006: PRINT " PAARUNG
      EN NACH CH-SYSTEM WEGEN VORGEBENER KONSTELL
      ATION NICHT MOGLICH! ": GET KEY XT$: GOTO 2
      0602                                               <D1T>
20616 DO                                                 <FTR>
20617 GOSUB 20701                                        <SQI>
20618 IF CR%=1 THEN EXIT                                 <1B1>
20619 PN%=PA%+1: GOSUB 20111                             <I5R>
20620 WINDOW 0,5,79,20: CHAR ,0,(MM%-5),CR$: GOSU
      B 20121: PRINT : MM%=PEEK(235)                     <3KH>
20621 PRINT "(2HOME)"                                    <K2E>
20628 IF GP%=MS% THEN EXIT                               <17Q>
20629 LOOP                                               <SF1>
20630 IF GP%<MS% THEN 20602                              <37Q>
20631 RETURN                                             <937>
20701 CR%=1: EY%=21: EL%=3: CHAR ,0,21,CR$: CHAR
      ,0,23,CL$                                          <I1R>
20702 EX%=5: E$(0)="": E$(3)=ZA$: E$(2)=""               <EN4>
20703 GOSUB 3001: IF XT$=CHR$(27) THEN 20817             <MU7>
20705 S0%=VAL(E$(1)): IF S0%>MS% THEN GOSUB 2006:
      GOTO 20703                                         <6BU>
20706 IF S0%>0 THEN 20740                                <1EP>
20707 GOSUB 20370: IF CR%=1 OR E$(1)="" THEN 2070
      1                                                  <D94>
20710 IJ=1                                               <TB5>
20715 DO WHILE IJ<=MS%                                   <AQD>
20716 IF LEFT$(NA$(IJ,1)+", "+NA$(IJ,2),LEN(E$(1)
      ))<>E$(1) THEN 20730                               <LC4>
20720 GOSUB 20820: IF CA%=0 THEN S0%=IJ: EXIT            <PSM>
20730 CA%=0: IJ=IJ+1: LOOP                               <MIJ>
20731 IF S0%=0 THEN GOSUB 2006: GOTO 20701               <N43>
20740 IJ=S0%: GOSUB 20820: IF CA%=1 THEN CA%=0: G
      OSUB 2006: GOTO 20701                              <N3D>
20774 XB$=NA$(S0%,1): IF NA$(S0%,2)<>"" THEN XB$=
      XB$+", "+NA$(S0%,2)                                <LV2>
20775 CHAR ,0,21,CR$: PRINT TAB(5) RIGHT$("  "+STR
      $(S0%),3);TAB(9) LEFT$(XB$,28);                    <4BD>
20776 CR%=1: EX%=38: EL%=3: E$(0)="": E$(3)=ZA$:
      E$(2)="(SYNTH.:145)SZ": CHAR ,38,21,CR$            <SG2>
20777 CHAR ,0,23,CL$: PRINT " (RVSON) Z (RVOFF) -
      ZURUCKGEZOGEN(2SPACE)*(3SPACE,RVSON) S (RV
      OFF) - SPIELFREI                                   <7UE>
20778 GOSUB 3001: IF XT$=CHR$(27) THEN 20817             <NM2>
20779 IF XT$="(SYNTH.:145)" THEN 20701                   <PBK>
20780 IF XT$="S" THEN S1%=255: GOTO 20812                <5AV>
20790 IF XT$="Z" THEN S1%=0: EG%=9: GOTO 20816           <F63>
20800 S1%=VAL(E$(1)): IF S1%>MS% THEN GOSUB 2006:
      GOTO 20776                                         <IKL>
20801 IF S1%>0 THEN 20811                                <2UV>
20802 GOSUB 20395: IF CR%=1 OR E$(1)="" THEN 2077
      6                                                  <HHG>
20803 IJ=1                                               <74D>
20804 DO WHILE IJ<=MS%                                   <J6A>
20805 IF IJ=S0% THEN 20809                               <DV1>
20806 IF LEFT$(NA$(IJ,1)+", "+NA$(IJ,2),LEN(E$(1)
      ))<>E$(1) THEN 20809                               <FCS>
20807 GOSUB 20820: IF CA%=1 THEN 20809                   <A6H>
20808 GOSUB 20821: IF CA%=0 THEN S1%=IJ: EXIT            <5N6>
20809 CA%=0: IJ=IJ+1: LOOP                               <F1L>
20810 IF S1%=0 THEN GOSUB 2006: GOTO 20776               <VB0>
20811 IF S0%=S1% THEN GOSUB 2006: GOTO 20776             <2SI>
20812 IJ=S1%: GOSUB 20820: IF CA%=1 THEN CA%=0: G
      OSUB 2006: GOTO 20776                              <NR8>
20813 IF DR%<RD% OR DR%<2 THEN 20815                     <JQS>
20814 IF DR%=RD% AND DR%>1 THEN IJ=S1%: GOSUB 208
      21: IF CA%=1 THEN CA%=0: GOSUB 2006: GOTO 2
      0776                                               <01H>
20815 EG%=0: IF S1%=255 THEN EG%=8                       <B69>
20816 SP%=S0%: SO%=S1%: CR%=0                            <O9H>
20817 CHAR ,0,21,CR$: RETURN                             <AGL>
20820 BANK 1: POKE DEC("3820"),IJ: SYS UP%: CA%=S
      GN(PEEK(DEC("3820"))): RETURN                      <F6B>
20821 BANK 1: POKE 16940,S0%: POKE 16941,IJ: SYS
      16942: CA%=PEEK(DEC("3820")): RETURN               <QDQ>
20830 M1$=" / PAARUNG AUFHEBEN": GOSUB 10011             <HHI>
20831 CHAR ,0,23,CL$                                     <TVI>
20832 PRINT " GEBEN SIE DIE NUMMER DER AUFZUHEBEN
      DEN PAARUNG EIN!": PRINT " (RVSON) A (RVOFF
      ) - ALLE PAARUNGEN!";                              <S8F>
20833 CHAR ,0,21,CR$: EY%=21: EX%=1: EL%=3: E$(0)
      ="": E$(2)="A": E$(3)=ZA$                          <TTR>
20834 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN            <MR6>
20835 IF XT$="A" THEN PN%=256: GOTO 20837                <QG4>
20836 PN%=VAL(E$(1)): IF PN%=0 OR PN%>PA% THEN GO
      SUB 2006: GOTO 20834                               <GVC>
20837 GOSUB 20101: CHAR ,0,21,CR$: IF PN%<256 THE
      N GOSUB 20121                                      <UT1>
20838 CHAR ,0,23,CL$: PRINT " PAARUNG(EN) AUFHEBE
      N? "JN$                                            <QFM>
20839 GET KEY XT$: IF XT$<>"J" THEN 20831                <TEA>
20840 BANK 1: IF PN%=256 THEN PA%=0: GP%=0: SE%=0
      : GOTO 20847                                       <RTR>
20842 PA%=PA%-1: IF EG%>0 THEN SE%=SE%-1                 <HJ7>
20843 GP%=GP%-1: IF SO%<>0 AND SO%<>255 THEN GP%=
      GP%-1                                              <KCR>
20844 FOR IJ=PN% TO PA%: POKE DEC("3000")+IJ,PEEK
      (DEC("3001")+IJ)                                   <9VG>
20845 POKE DEC("3100")+IJ,PEEK(DEC("3101")+IJ)           <LTQ>
20846 POKE DEC("3200")+IJ,PEEK(DEC("3201")+IJ): N
      EXT IJ                                             <QOM>
```

Listing 1. »CH-VERWALTUNG« (Fortsetzung)

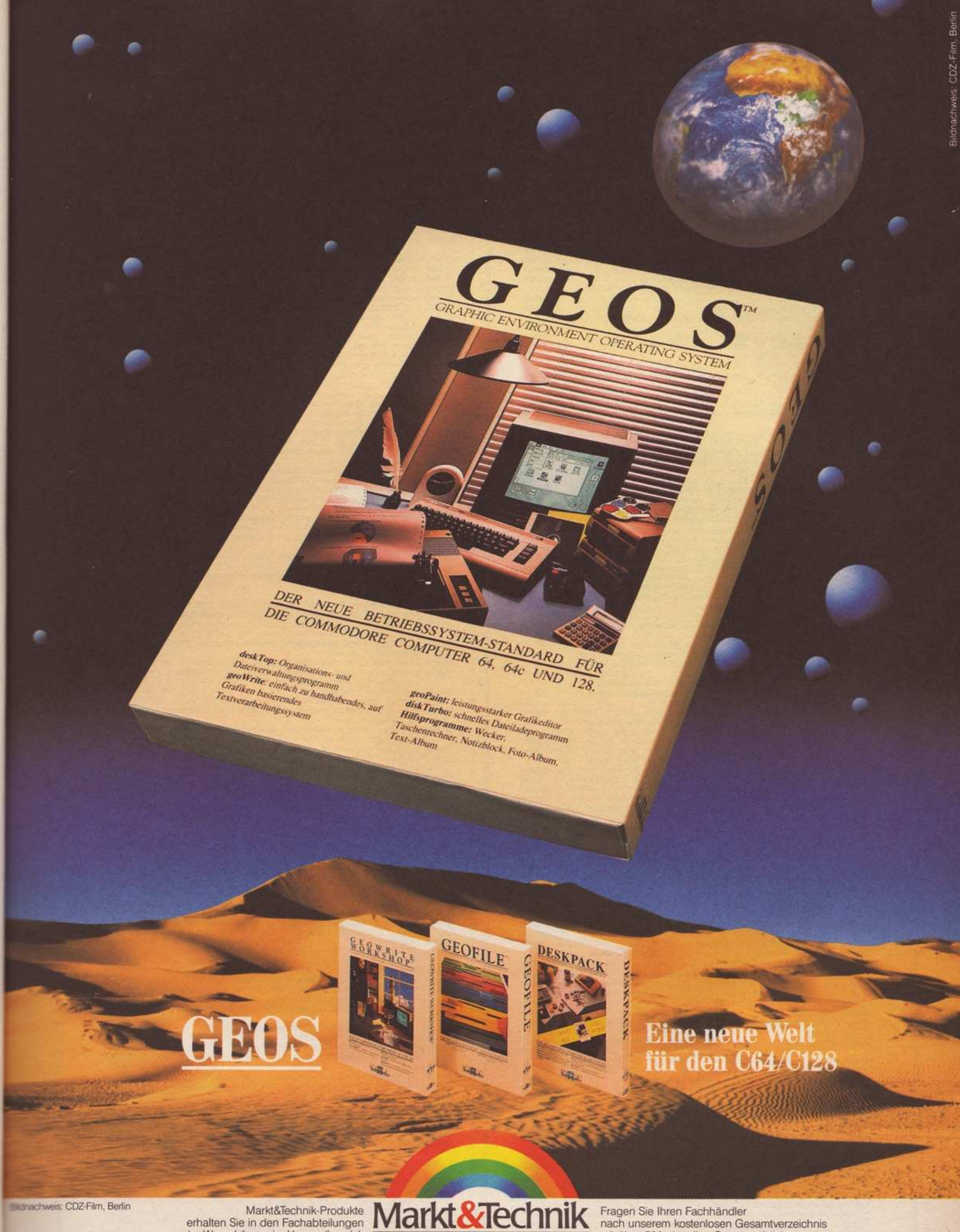
```
20847 POKE DEC("3000"),PA%: POKE DEC("3100"),GP%:
       POKE DEC("3200"),SE%                                <OE6>
20849 CX%=1: GOSUB 20201: CX%=0: BANK 1: SYS GT%:
       IF PA%>0 THEN 20831: ELSE RETURN                     <IV5>
20901 IF PA%=0 THEN RETURN                                 <Q58>
20902 M1$=" / ERGEBNISNEUEINGABE": GOSUB 10011             <0M6>
20903 CHAR ,0,23,CL$: PRINT XE$: PRINT YE$                 <Q6N>
20904 PN%=0: CA%=0: CR%=0                                  <D3I>
20905 DO                                                   <HE1>
20906 BANK 1: POKE DEC("3821"),CA%: POKE DEC("445
      2"),PN%: SYS DEC("443D")                             <50C>
20907 CR%=PEEK(DEC("3820")): IF CR%=1 THEN EXIT            <6SC>
20908 PN%=PEEK(DEC("4452")): GOSUB 20101                   <LE1>
20910 CA%=0: CHAR ,0,21,CR$: GOSUB 20121: PRINT C
      HR$(27)"J";TAB(71);: BANK 15: SYS 52591: GE
      T KEY XT$: SYS 52639: CHAR ,0,21,CR$: IF XT
      $=CHR$(27) THEN EXIT                                 <3DF>
20911 IF XT$="-" THEN CA%=1: GOTO 20929                    <DCH>
20919 IF INSTR("01234567",XT$)=0 THEN 20929                <2GF>
20920 IF EG%=4 THEN SE%=SE%-1                              <IK3>
20921 EG%=VAL(XT$): BANK 1: GOSUB 20115                    <UI9>
20922 GOSUB 21235: GOTO 20929                              <CE3>
20929 LOOP                                                 <401>
20930 CA%=0: IF CR%=1 THEN CHAR ,0,23,CL$: PRINT
      " ALLE ERGEBNISSE EINGETRAGEN!": GOSUB 2001
      : GET KEY XT$                                        <VCB>
20931 RETURN                                               <8V7>
20951 MO%=1: IF GP%<MS% THEN RETURN                        <VHI>
20952 MO%=2: IF SE%<PA% THEN RETURN                        <2L5>
20953 MO%=3                                                <I3A>
20954 BANK 1: SYS DEC("43D3"): IF PEEK(DEC("3820"
      ))=1 THEN MO%=4                                      <7E1>
20958 RETURN                                               <0Q7>
21001 GOSUB 20951                                          <GIB>
21002 M1$="": GOSUB 10011                                  <KIN>
21003 CA%=0: CHAR ,0,23,CL$: PRINT " (RVSON) P (R
      VOFF) - PAARUNGEN(2SPACE)*(2SPACE,RVSON) E
      (RVOFF) - ERGEBNISSE(2SPACE)*(2SPACE,RVSON)
       N (RVOFF) - NFCHSTE RUNDE(2SPACE)*(2SPACE,
      RVSON) R (RVOFF) - ANDERE"                           <8OQ>
21004 PRINT " RUNDE(2SPACE)*(2SPACE,RVSON) T (RVO
      FF) - TABELLEN(2SPACE)*(2SPACE,RVSON) S (RV
      OFF) - SPIELERVERWALTUNG(2SPACE)*(2SPACE,RV
      SON) L (RVOFF) - LEGENDE FNDERN";: GOTO 210
      16                                                   <V0H>
21005 CHAR ,0,23,CL$: PRINT " (RVSON) L (RVOFF) -
       LISTE(2SPACE)*(2SPACE,RVSON) S (RVOFF) - A
      BSPEICHERN(2SPACE)*(2SPACE,RVSON) Z (RVOFF)
       - SPIELER FXR KOMMENDE RUNDEN"                      <7V2>
21006 PRINT " ZURXCKZIEHEN(2SPACE)*(2SPACE,RVSON)
       E (RVOFF) - PAARUNGSETIKETTEN(2SPACE)*(2SP
      ACE,RVSON) L (RVOFF) - LEGENDE FNDERN";: GO
      TO 21016                                             <RJB>
21007 CHAR ,0,23,CL$: PRINT " (RVSON) C (RVOFF) -
       CH-SYSTEM-KARTE(2SPACE)*(2SPACE,RVSON) F (
      RVOFF) - FUNKTIONSTASTEN BELEGEN(2SPACE)*(2
      SPACE,RVSON) L (RVOFF) - LEGENDE"                    <GKH>
21010 PRINT " FNDERN":                                     <QS1>
21016 YT$="ETSPEEL SLC": IF DR%=RD% AND RD%<MR% TH
      EN YT$=YT$+"Z"                                       <6CO>
21017 IF MO%>2 AND RD%=DR% AND RD%<MR% THEN YT$=Y
      T$+"N"                                               <FO7>
21018 IF RD%>1 THEN YT$=YT$+"R"                            <L9G>
21019 GET KEY XT$: IF XT$="C" THEN GOSUB 23001: G
      OTO 20015                                            <EU4>
21020 IF XT$=CHR$(27) THEN CX%=0: GOTO 21060               <QUD>
21027 IF INSTR(YT$,XT$)=0 THEN GOSUB 2006: GOTO 2
      1019                                                 <KVK>
21028 IF XT$="E" THEN GOSUB 21081: GOTO 21001              <9RD>
21029 IF XT$="Z" THEN GOSUB 21501: GOTO 21001              <CFC>
21030 IF XT$="L" THEN CA%=CA%+1: ON CA% GOTO 2100
      5,21007: GOTO 21003                                  <017>
21031 IF XT$="P" THEN GOSUB 21101: GOTO 21001              <4F6>
21032 IF XT$="E" THEN GOSUB 21201: GOTO 21001              <7V5>
21033 IF XT$="T" THEN GOSUB 22001: GOTO 20015              <A73>
21034 IF XT$="S" THEN CX%=1: GOTO 21060                    <A3N>
21035 IF XT$="F" THEN GOSUB 53001: GOTO 20015              <J75>
21036 IF XT$="R" THEN 21040                                <IIT>
21037 IF XT$="N" THEN 21053                                <S6V>
21038 IF XT$="L" THEN GOSUB 20201: GOTO 21002              <3BE>
21039 IF XT$="S" THEN GOSUB 21070: GOSUB 51011: G
      OTO 21003                                            <B1C>
21040 GOSUB 21070                                          <H6V>
21042 CHAR ,0,23,CL$                                       <9FG>
21043 PRINT " GEBEN SIE DIE GEWXNSCHTE RUNDENNUMM
      ER EIN:": PRINT " (RVSON) A (RVOFF) - RUNDE
      , IN DER NOCH ERGEBNISSE ODER PAARUNGEN OFF
      ENSTEHEN";                                           <UV9>
21044 EY%=23: EL%=2: EX%=44: E$(0)="": E$(2)="A":
       E$(3)=ZA$                                           <N52>
21045 GOSUB 3001: IF XT$=CHR$(27) THEN 21003               <LMA>
21046 IF XT$="A" THEN BEGIN                                <3LT>
21047 BANK 1: DR%=1: DO WHILE DR%<RD%: IF PEEK(DE
      C("1000")+DR%)<4 THEN EXIT                           <0PJ>
21048 DR%=DR%+1: LOOP                                      <D3B>
21049 BEND : GOTO 21052                                    <TFB>
21050 IF VAL(E$(1))=0 OR VAL(E$(1))>RD% THEN GOSU
      B 2006: GOTO 21045                                   <02V>
21051 DR%=VAL(E$(1))                                       <T6F>
21052 BANK 1: POKE DEC("1000"),DR%: GOTO 20011             <TTF>
21053 GOSUB 21070                                          <HMV>
21055 RD%=RD%+1: BANK 1: POKE DEC("1000")+RD%,1:
      DR%=RD%: SYS LE%: POKE DEC("1000"),DR%: TU%
      =2: GOTO 20011                                       <K14>
21060 GOSUB 21070: CHAR ,0,23,CL$: PRINT DL$;JN$:
       GET KEY XT$: IF XT$="N" THEN 21067                  <4C6>
21061 IF XT$=CHR$(27) THEN CX%=0: GOTO 21001               <P22>
21063 GOSUB 51011: IF CR%=1 THEN 21001                     <VSU>
21067 ON CX% GOTO 40001: GOTO 63901                        <RCF>
21070 BANK 1: SYS GT%: POKE DEC("1000")+DR%,MO%:
      POKE DEC("1000"),DR%: RETURN                         <TKS>
21081 GOSUB 5001: IF CR%=1 THEN RETURN                     <FBU>
21082 OPEN 4,4: OPEN 5,5                                   <1H9>
21083 PN%=1: DO WHILE PN%<=PA%                             <EGH>
21084 GOSUB 20101: IF EG%>0 THEN 21094                     <AFB>
21085 PRINT#5,CHR$(27)"@";: PRINT#5,CHR$(27)"!(DO
      WN)";                                                <1HC>
21086 PRINT#4                                              <HM1>
21088 XX$="(3SPACE)BRETT"+STR$(PN%): GOSUB 21096            <VJS>
21089 PRINT#4: XX$="(3SPACE)WEIT:(4SPACE)"+RIGHT$
      (" "+STR$(SP%),3)+"(2SPACE)"+NA$(SP%,1): IF
       NA$(SP%,2)<>"" THEN XX$=XX$+", "+NA$(SP%,2
      )                                                    <DAL>
21090 GOSUB 21096                                          <VEI>
21091 XX$="(3SPACE)SCHWARZ: "+RIGHT$(" "+STR$(SO%
      ),3)+"(2SPACE)"+NA$(SO%,1): IF NA$(SO%,2)<>
      "" THEN XX$=XX$+", "+NA$(SO%,2)                      <UF2>
21092 GOSUB 21096                                          <UEI>
21093 PRINT#4: XX$="(3SPACE)ERGEBNIS:(5SPACE):":
      GOSUB 21096: PRINT#4: PRINT#4                        <NHP>
21094 PN%=PN%+1: LOOP                                      <JBB>
21095 CLOSE 4: CLOSE 5: RETURN                             <8CQ>
21096 XX$=LEFT$(XX$+SP$,48)                                <HAQ>
21097 PRINT#4,XX$;XX$: RETURN                              <745>
21101 M1$=" / PAARUNGEN": GOSUB 10011                      <D73>
21102 CHAR ,0,23,CL$                                       <PBG>
21103 PRINT " (RVSON) N (RVOFF) - PAARUNGSNEUEING
      ABE(2SPACE)*(2SPACE,RVSON) A (RVOFF) - PAAR
      UNGEN AUFHEBEN(2SPACE,RVSON) L (RVOFF) - LI
      STE"                                                 <6ET>
21104 PRINT " (RVSON) F (RVOFF) - SPIELER SUCHEN(
      2SPACE)*(2SPACE,RVSON) P (RVOFF) - NOCH NIC
      HT GEPAARTE SPIELER";                                <R10>
21105 YT$=CHR$(27)+CHR$(27)+"LFP"                          <7N4>
21106 IF GP%<MS% THEN YT$="N"+MID$(YT$,2,20)               <9FQ>
21107 IF PA%>0 THEN YT$=LEFT$(YT$,1)+"A"+MID$(YT$
      ,3,20)                                               <7HV>
21108 CHAR 1,0,21,CR$: GET KEY XT$                         <GLQ>
21109 XT%=INSTR(YT$,XT$): IF XT$=CHR$(27) OR XT%=
      0 THEN RETURN                                        <I7T>
21110 ON XT% GOSUB 20601,20830,20201,20260,20420           <1F2>
21111 IF GP%=MS% THEN RETURN                               <P4S>
21112 GOTO 21101                                           <BTC>
21201 M1$=" / ERGEBNISSE": GOSUB 10011                     <SFE>
21202 CHAR ,0,23,CL$: PRINT " (RVSON) N (RVOFF) -
       ERGEBNISNEUEINGABE(2SPACE)*(2SPACE,RVSON)
      L (RVOFF) - ALLE ERGEBNISSE LTSCHEN(2SPACE)
      *(2SPACE,RVSON) E (RVOFF) - OFFEN-"                  <506>
21203 PRINT " STEHENDE ERGEBNISSE(2SPACE)*(2SPACE
      ,RVSON) L (RVOFF) - LISTE(2SPACE)*(2SPACE,R
      VSON) F (RVOFF) - SPIELER SUCHEN";                   <GM6>
21204 CHAR ,0,21,CR$: EY%=21: EX%=1: EL%=3: E$(0)
      ="": E$(2)="LFNEL": E$(3)=ZA$: GOSUB 3001:
      IF XT$=CHR$(27) THEN RETURN                          <NQO>
21205 IF XT$="L" THEN GOSUB 21260: GOTO 21201              <0B8>
21206 IF XT$="E" THEN GOSUB 21250: GOTO 21201              <1VB>
21207 IF XT$="N" THEN GOSUB 20901: GOTO 21201              <KVA>
21209 IF XT$="L" THEN GOSUB 20201: GOTO 21201              <1V4>
21210 IF XT$="F" THEN GOSUB 20260: GOTO 21201              <77F>
21211 PN%=VAL(E$(1)): IF PN%>PA% THEN GOSUB 2006:
       GOTO 21204                                          <GGH>
21212 IF PN%=0 THEN GOSUB 21300: IF CR%=1 THEN 21
      202                                                  <HHA>
21220 GOSUB 20101: IF EG%>7 THEN GOSUB 2006: GOTO
       21204                                               <OJ2>
21230 CHAR ,0,23,CL$: PRINT XE$: PRINT YE$: CHAR
      ,0,21,CR$: GOSUB 20121                               <JGD>
21231 PRINT CHR$(27)"J";TAB(71): BANK 15: SYS 525
      91: GET KEY XT$: SYS 52639                           <KHM>
21232 IF INSTR("01234567",XT$)=0 THEN 21202                <IEC>
21233 IF EG%>0 THEN SE%=SE%-1                              <GTB>
21234 EG%=VAL(XT$): BANK 1: GOSUB 20115: GOSUB 21
      235: GOTO 21202                                      <V3M>
21235 IX=5: DO WHILE IX<20                                 <B7F>
21236 OPEN 3,3: CHAR ,1,IX: YT$="": FOR IK=1 TO 3
      : GET #3,XT$: YT$=YT$+XT$: NEXT IK: CLOSE 3          <840>
21237 IF VAL(YT$)=PN% THEN CHAR ,0,(IX),CR$: GOSU
      B 20121: EXIT                                        <16A>
21238 IX=IX+1: LOOP                                        <57P>
21239 IF IX<20 THEN RETURN                                 <140>
21240 WINDOW 0,5,79,20: CHAR ,0,(MM%-5),CR$: GOSU
      B 20121: PRINT : MM%=PEEK(235): PRINT "(2HO
      ME)": RETURN                                         <KA9>
21250 M1$=" / OFFENSTEHENDE ERGEBNISSE": GOSUB 10
      011                                                  <NS6>
21251 BANK 1: SYS DEC("4490"): SR%(0)=PEEK(DEC("0
      800")): IF SR%(0)=0 THEN CR%=1: GOTO 20930           <LTI>
21252 FOR IJ=1 TO SR%(0): SR%(IJ)=PEEK(DEC("0800"
      )+IJ): NEXT IJ: IJ=1: GOTO 20202                     <85H>
21260 CHAR ,0,23,CL$: PRINT " ALLE ERGEBNISSE LTS
      CHEN? "JN$: GET KEY XT$: IF XT$<>"J" THEN R
      ETURN                                                <E55>
21261 BANK 1: SYS DEC("4477"): SE%=PEEK(DEC("3200
      ")): GOTO 20201                                      <LIA>
21300 CHAR ,0,23,CL$                                       <HFG>
21301 CHAR ,0,21,CR$: CR%=0: EY%=21: EX%=5: EL%=3
      : E$(3)=ZA$: E$(2)=""                                <ODS>
21303 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: RET
      URN                                                  <F6B>
21304 IJ=VAL(E$(1)): IF IJ>MS% THEN GOSUB 2006: G
      OTO 21303                                            <Q93>
21305 IF IJ=0 THEN GOSUB 21400: IF CR%=1 THEN 213
      01                                                   <UBK>
21306 BANK 1: POKE DEC("3820"),IJ: SYS DEC("43EC"
      )                                                    <67S>
21307 PN%=PEEK(DEC("3820"))                                <SK3>
21311 IF PN%=0 THEN GOSUB 2006: GOTO 21301                 <8SC>
21312 RETURN                                               <HO7>
21400 CHAR ,0,21,CR$: CR%=0: EY%=21: EX%=9: EL%=2
      8: E$(3)=ZA$+BS$+",": E$(2)=""                       <ME4>
21401 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: RET
      URN                                                  <CUB>
21402 IJ=MS%: DO WHILE IJ>0                                <034>
21403 IF E$(1)=LEFT$(NA$(IJ,1)+", "+NA$(IJ,2),LEN
      (E$(1))) THEN EXIT                                   <NT2>
21404 IJ=IJ-1: LOOP                                        <P1P>
21405 IF IJ=0 THEN GOSUB 2006: GOTO 21401                  <B32>
21406 RETURN                                               <127>
21501 CHAR ,0,23,CL$                                       <L3H>
21510 PRINT " GEBEN SIE DEN SPIELER EIN, DER ZURX
      CKGEZOGEN WERDEN SOLL!"                              <OTS>
21515 CHAR ,0,21,CR$: EY%=21: EL%=3: EX%=5: E$(0)
      ="": E$(2)="": E$(3)=ZA$                             <7E3>
21520 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN              <BOI>
21525 SP%=VAL(E$(1)): IF SP%=0 THEN 21540                  <9AB>
21530 IF SP%>MS% THEN GOSUB 2006: GOTO 21520               <EOB>
21535 GOTO 21570                                           <CP3>
21540 GOSUB 20370: IF CR%=1 THEN CR%=0: GOTO 2151
      5                                                    <38Q>
21545 SP%=MS%                                              <897>
21550 DO WHILE SP%>0                                       <PDK>
21555 IF E$(1)=LEFT$(NA$(SP%,1)+", "+NA$(SP%,2),L
      EN(E$(1))) THEN EXIT                                 <JIN>
```

```
21560 SP%=SP%-1: LOOP                                   <05A>
21565 IF SP%=0 THEN GOSUB 2006: GOTO 21540             <MB6>
21570 SO%=0: EG%=0: CHAR ,0,21,CR$: GOSUB 20122        <707>
21575 CHAR ,0,23,CL$                                   <T7H>
21580 PRINT " ZUR¥CKZIEHEN? "JN$                       <A3E>
21585 GET KEY XT$: XT%=INSTR("JN",XT$): IF XT%=0
      THEN 21585                                       <20J>
21595 BANK 1: FOR II=RD%+1 TO MR%                      <I12>
21596 ON XT% GOSUB 21600,21602                         <BNP>
21597 NEXT II: GOTO 21501                              <5D7>
21600 POKE DEC("2000")+16*SP%+II,16: RETURN            <2KB>
21602 POKE DEC("2000")+16*SP%+II,0 : RETURN            <OH5>
22001 US$="###.#": M1$=" / TABELLEN": GOSUB 10011      <F77>
22004 CHAR ,0,3,CL$: PRINT "(RVSON) PLZ ¥SPIELER
      (NR./TITEL/NAME)(23SPACE)¥PUNKTE(4SPACE)¥BH
      OLZ ¥MBHOLZ(2SPACE)"                             <AR7>
22005 CHAR ,0,22,"++++++++++++++++++++++++++++++++++
      ++++++++++++++++++++++++++++++++++++++++
      ++++++"                                          <CB3>
22010 PRINT " (RVSON) F (RVOFF) - FORTSCHRITTSTAB
      ELLE AUF DRUCKER(2SPACE)*(2SPACE,RVSON) T (
      RVOFF) - RANGTABELLE OHNE ..."                   <CN2>
22015 PRINT " (RVSON) I (RVOFF) - RANGTABELLE MIT
       BUCHOLZWERTUNG";                                <RD5>
22020 YT$="FTI"                                        <BDN>
22025 GET KEY XT$: IF INSTR(YT$,XT$)=0 THEN RETUR
      N                                                <SOF>
22030 ON INSTR(YT$,XT$) GOSUB 19001,22100,22200        <B6T>
22035 GOTO 22001                                       <FL2>
22100 BANK 1: SYS GT%: POKE DEC("3300"),MS%: POKE
       DEC("1000"),DR%+1: SYS 17017: POKE DEC("10
      00"),DR%: IJ=1                                   <MJQ>
22110 WINDOW 0,5,79,20,1: IK=IJ                        <UHT>
22115 DO WHILE IJ<=MS%                                 <AIC>
22120 IL=IJ: GOSUB 22300                               <BUT>
22125 PRINT : IF IJ=IK+14 THEN EXIT                    <IGJ>
22130 IJ=IJ+1: LOOP : IJ=IK+14                         <PGJ>
22135 PRINT "(2HOME)": CHAR ,0,23,CL$: PRINT " (R
      VSON) D (RVOFF) - TABELLE AUF DRUCKER": PRI
      NT BX$;                                          <V6L>
22140 PRINT "(2HOME)": GET KEY XT$: IF INSTR("+-D
      (HOME,CLR)PP",XT$)=0 THEN RETURN                 <F0U>
22142 IF XT$="(CLR)" AND MS%>14 THEN IJ=MS%-14: G
      OTO 22110                                        <351>
22145 IF XT$="D" THEN GOSUB 37001: GOTO 22135          <L7V>
22146 IF XT$="(HOME)" THEN IJ=1: GOTO 22110            <7CD>
22150 IF XT$="+" AND IJ<MS% THEN IJ=IJ+1: IK=IK+1
      : IL=IJ: WINDOW 0,5,79,20: CHAR ,0,15,CR$:
      GOSUB 22300: PRINT : GOTO 22140                  <MUS>
22155 IF XT$="-" AND IK>1 THEN IJ=IJ-1: IK=IK-1:
      IL=IK: WINDOW 0,5,79,20: CHAR ,0,0,CHR$(27)
      +"I"+CR$: GOSUB 22300: GOTO 22140                <NV8>
22156 IF XT$="P" AND IJ<MS% THEN IJ=IJ+1: GOTO 22
      110                                              <GG3>
22157 IF XT$="P" AND IK-15>0 THEN IJ=IK-15: GOTO
      22110                                            <4Q9>
22160 GOTO 22140                                       <CP4>
22200 BANK 1: SYS GT%: POKE DEC("3300"),MS%: POKE
       DEC("1000"),DR%+1: SYS TA%: POKE DEC("1000
      "),DR%: IJ=1                                     <FEB>
22205 GOTO 22110                                       <CP1>
22300 BANK 1: SP%=PEEK(DEC("3300")+IL)                 <8OQ>
22305 P1=PEEK(DEC("3400")+SP%)/2                       <AEG>
22310 H1=PEEK(DEC("3500")+SP%)                         <ORP>
22315 B1=(PEEK(DEC("3600")+SP%)+PEEK(DEC("0600")+
      SP%)*256)/2                                      <VVN>
22320 A1=(PEEK(DEC("3700")+SP%)+PEEK(DEC("0700")+
      SP%)*256)/2                                      <307>
22325 PRINT " " RIGHT$("(2SPACE)"+STR$(IL),3)".";      <SIL>
22330 PRINT TAB(6)"(" RIGHT$("(2SPACE)"+STR$(SP%)
      ,3)") "NA$(SP%,0);                               <N1R>
22335 PRINT TAB(16)NA$(SP%,1);: IF NA$(SP%,2)<>""
       THEN PRINT ", "NA$(SP%,2);                      <UES>
22340 PRINT TAB(54);: PRINT USING US$;P1;              <HTV>
22342 IF H1>0 THEN PRINT " +"+RIGHT$(STR$(H1),2)"
      H";                                              <BHL>
22345 IF B1>0 AND DR%>1 THEN PRINT TAB(66);: PRIN
      T USING US$;B1;                                  <DAP>
22350 IF A1>0 AND DR%>2 THEN PRINT TAB(73);: PRIN
      T USING US$;A1;                                  <O3C>
22355 BANK 15: SYS 2848: RETURN                        <V10>
22400 CHAR ,0,3,CR$: PRINT "(RVSON) PLZ ¥SPIELER
      (NR./TITEL/NAME)(23SPACE)¥PUNKTE(4SPACE)¥BH
      OLZ ¥MBHOLZ(2SPACE)";: RETURN                    <NJM>
23001 M1$=" / CH-SYSTEM-KARTE(N)": GOSUB 10011         <N1Q>
23002 CHAR ,0,3,CL$: BANK 1: POKE DEC("3300"),MS%
      : SYS GT%                                        <3UP>
23003 PRINT "(RVSON) NR.¥(TITEL) NAME, VORNAME(18
      SPACE)¥VEREIN (NATION/STADT)(3SPACE)¥INGO¥E
      LO(2SPACE)"                                      <5U5>
23005 CHAR ,0,7,CL$                                    <4K0>
23006 PRINT "(RVSON) RD.¥GEGNER: (RANGNR./NAME/NA
      TION)(13SPACE)(INGO/ELO)¥FA.¥ERGEBNIS¥E PUN
      KTE "                                            <MMT>
23007 CHAR ,0,22,"++++++++++++++++++++++++++++++++++
      ++++++++++++++++++++++++++++++++++++++++
      ++++++"                                          <OB3>
23010 CHAR ,0,23,CL$: PRINT " GEBEN SIE DIE RANGN
      UMMER ODER DEN NAMEN DES GEW¥NSCHTEN SPIELE
      R   N! "                                         <EPQ>
23011 P[illegible]" (RVSON) D (RVOFF) - CH-SYSTEM-KARTE
      (N) AUF DRUCKER";                                <PB6>
23012 CHAR ,0,5,CR$: WINDOW 0,9,79,21,1: PRINT "(
      2HOME)": EY%=5: EX%=1: EL%=3: E$(0)="": E$(
      2)="D": E$(3)=ZA$                                <TA4>
23013 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN          <BBG>
23014 IF XT$="D" THEN GOSUB 39001: GOTO 23010          <DFU>
23015 SP%=VAL(E$(1)): IF SP%>MS% THEN GOSUB 2006:
       GOTO 23013                                      <7KP>
23016 IF SP%>0 THEN 23025                              <0QV>
23017 CHAR ,0,24,CR$: CHAR ,0,5,CR$                    <UHK>
23018 EX%=9: EL%=35: E$(0)="": E$(2)="(SYNTH.:145
      )": E$(3)=","+ZA$+BS$                            <2RT>
23019 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN          <EB4>
23020 IF XT$="(SYNTH.:145)" THEN 23010                 <Q7P>
23021 CR%=0: SP%=1: DO WHILE SP%<=MS%                  <SS1>
23022 IF E$(1)=LEFT$(NA$(SP%,1)+", "+NA$(SP%,2),L
      EN(E$(1))) THEN EXIT                             <9IN>
23023 SP%=SP%+1: LOOP                                  <8FB>
23024 IF SP%>MS% THEN GOSUB 2006: GOTO 23017           <E48>
23025 CHAR ,0,5,CR$: CA%=1: GOSUB 40101: WINDOW 0
      ,9,79,21,1: GOSUB 23026: GOTO 23060              <62H>
23026 BANK 1: P1=0: H1=0: FOR IJ=1 TO RD%              <BU4>
23028 SO%=PEEK(DEC("1000")+SP%*16+IJ)                  <300>
23029 IK=PEEK(DEC("2000")+SP%*16+IJ): P0=(IK AND
      3): H0=(IK AND 4)/4: FA%=IK AND 192: IL=IK
      AND 15: GS%=IK AND 16: EG%=0: IF GS%=0 THEN
       23043                                           <R7J>
23034 IF IL=0 THEN EG%=2: GOTO 23043                   <IDN>
23035 IF IL=1 THEN EG%=3: GOTO 23043                   <JLJ>
23036 IF IL=2 THEN EG%=1: GOTO 23043                   <HDV>
23037 IF IL=4 THEN EG%=4: GOTO 23043                   <LL7>
23038 IF IL=10 THEN BEGIN                              <IPB>
23039 IF SO%=255 THEN EG%=8: ELSE EG%=5                <2OP>
23040 BEND : GOTO 23043                                <NR2>
23041 EG%=6: IF(PEEK(DEC("2000")+SO%*16+IJ) AND 3
      )=0 THEN EG%=7                                   <E16>
23043 P1=P1+P0: H1=H1+H0                               <A3M>
23052 IF SO%=0 AND GS%=0 THEN SO%=256                  <7LC>
23053 IF SO%=0 AND GS%>0 THEN EG%=9                    <FTL>
23057 SR%(15+IJ)=SGN(FA%-128)+1                        <M9E>
23059 SR%(IJ)=SO%: SR%(30+IJ)=EG%: SR%(45+IJ)=P1:
       SR%(60+IJ)=H1: NEXT IJ: RETURN                  <M0N>
23060 WINDOW 0,9,79,21,1                               <94T>
23061 IK=1: IJ=1: DO WHILE IJ<=RD%                     <S3T>
23062 IL=IJ: GOSUB 23070: PRINT : IF IJ=IK+11 THE
      N EXIT                                           <3BF>
23064 IJ=IJ+1: LOOP                                    <H7P>
23065 PRINT "(2HOME)": CHAR ,0,23,CL$: PRINT " (R
      VSON) + (RVOFF) / (RVSON) - (RVOFF) - BL¥TT
      ERN": PRINT " (RVSON) D (RVOFF) - CH-SYSTEM
      -KARTE AUF DRUCKER";: GET KEY XT$: IF INSTR
      ("D+-",XT$)=0 THEN 23010                         <47S>
23066 IF XT$="D" THEN IX=SP%: IY=SP%: GOSUB 39009
      : GOTO 23010                                     <QD0>
23067 IF XT$="+" AND IJ<RD% THEN IJ=IJ+1: IK=IK+1
      : IL=IJ: WINDOW 0,9,79,21: CHAR ,0,12,CR$:
      GOSUB 23070: PRINT : GOTO 23065                  <LOH>
23068 IF XT$="-" AND IK>1 THEN IK=IK-1: IJ=IJ-1:
      IL=IK: WINDOW 0,9,79,21: CHAR ,0,0,CHR$(27)
      +"I"+CR$: GOSUB 23070: GOTO 23065                <JJC>
23069 GOTO 23065                                       <C52>
23070 PRINT USING "###";IL;: GOSUB 23090               <B2I>
23071 IF SO%=0 THEN XX$="ZUR¥CKGEZOGEN": GOTO 230
      76                                               <MCF>
23072 IF SO%=256 THEN XX$="NOCH NICHT GEPAART": G
      OTO 23076                                        <KVA>
23073 IF SO%=255 THEN XX$="SPIELFREI": GOTO 23076      <VKP>
23074 XX$="("+RIGHT$(" "+STR$(SO%),3)+") "+LEFT$(
      NA$(SO%,0)+SP$,4)+NA$(SO%,1): IF NA$(SO%,2)
      <>"" THEN XX$=XX$+", "+NA$(SO%,2)                <06I>
23075 XX$=LEFT$(XX$+SP$,38)+" "+LEFT$(NA$(SO%,3),
      3)                                               <BLE>
23076 PRINT TAB(5)XX$;: IF SO%=0 OR SO%>254 THEN
      23079                                            <7BA>
23077 IF NA%(SO%,0)>0 THEN PRINT TAB(49);: PRINT
      USING "###";NA%(SO%,0);                          <N17>
23078 IF NA%(SO%,1)>0 THEN PRINT TAB(53);: PRINT
      USING "####";NA%(SO%,1);                         <JHA>
23079 PRINT TAB(59)FA$(FA%);                           <16S>
23081 PRINT TAB(62)EG$(EG%);TAB(71);: PRINT USING
       "##.#";P0;                                      <P56>
23082 IF H0>0 THEN PRINT " +";: PRINT USING "##";
      H0;: PRINT "H";                                  <1IK>
23083 RETURN                                           <9N7>
23090 SO%=SR%(IL): FA%=SR%(IL+15): EG%=SR%(IL+30)
      : P0=SR%(IL+45)/2: H0=SR%(IL+60): RETURN         <2DM>
30001 M0$="DIENST": M1$="": GOSUB 10011                <MGQ>
30005 CHAR ,0,3,CL$                                    <SL0>
30010 PRINT " (DOWN) (RVSON,2SPACE)1(2SPACE,RVOFF
      ) - FUNKTIONSTASTEN BELEGEN                      <KID>
30015 PRINT " (DOWN) (RVSON,2SPACE)2(2SPACE,RVOFF
      ) - INHALTSVERZEICHNIS DER DATEIDISKETTE         <RIP>
30020 PRINT " (DOWN) (RVSON,2SPACE)3(2SPACE,RVOFF
      ) - DISKETTE FORMATIEREN                         <168>
30025 PRINT " (DOWN) (RVSON,2SPACE)4(2SPACE,RVOFF
      ) - TURNIERDATEI L¥SCHEN                         <LVF>
30030 PRINT " (DOWN) (RVSON,2SPACE)5(2SPACE,RVOFF
      ) - TURNIERDATEI UMBENENNEN                      <MR7>
30035 PRINT " (DOWN) (RVSON,2SPACE)6(2SPACE,RVOFF
      ) - SONSTIGE DISKETTENBEFEHLE                    <620>
30040 PRINT " (DOWN) (RVSON,5SPACE,RVOFF) - ZUR¥C
      K INS HAUPTMENUE                                 <EOE>
30045 YT$="123456"                                     <1SL>
30050 GET KEY XT$: IF XT$=CHR$(27) THEN 63901          <NNC>
30055 XT%=INSTR(YT$,XT$): IF XT%=0 THEN 63901          <9SS>
30060 ON XT% GOSUB 53001,31001,32001,33001,34001,
      30100                                            <N7K>
30065 GOTO 30001                                       <D52>
30100 CHAR ,0,23,CL$: PRINT " DISKETTENBEFEHL:"        <8NO>
30101 EX%=18: EY%=23: EL%=40: E$(0)="": E$(2)="$@
      ": E$(3)=BS$+ZA$+":,;"                           <JJQ>
30105 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN          <F80>
30106 IF XT$="$" THEN WINDOW 1,4,79,21,1: DIRECTO
      RY : PRINT "(2HOME)": GOTO 30115                 <QCB>
30107 IF XT$="@" THEN 30115                            <KF2>
30108 IF E$(1)="" THEN RETURN                          <V8N>
30109 IF LEFT$(E$(1),2)="N:" THEN 32001                <UCT>
30110 OPEN 15,8,15: PRINT#15,E$(1)                     <VT9>
30115 CHAR ,0,23,CL$: PRINT " "DS$: CLOSE 15: GET
       KEY XT$: GOTO 30100                             <CF4>
31001 GOSUB 4101: IF CR%=1 THEN RETURN                 <9B6>
31005 NA$(0,0)=YD$: NA$(0,1)=XD$                       <FVJ>
31020 OPEN 15,8,15,"UJ": SLEEP 1: OPEN 2,8,2,"#":
       IJ=1: IL=1                                      <JB2>
31025 DO                                               <DDH>
31030 PRINT#15,"U1:2 0 18";IJ: IK=0: IF DS>0 THEN
       CR%=1: EXIT                                     <0E1>
31035 DO                                               <CDL>
31040 PRINT#15,"B-P 2";IK*32+2                         <BCD>
31045 GET #2,XD$: IF XD$<>CHR$(129) THEN 31090         <G5V>
31050 PRINT#15,"B-P 2";IK*32+5                         <BOE>
31055 GET #2,XD$: IF XD$<>"K" THEN 31090               <Q0Q>
31060 YD$=""                                           <AN9>
31065 DO                                               <9D1>
31070 GET #2,XD$: IF XD$=CHR$(160) THEN EXIT           <HL5>
31075 YD$=YD$+XD$                                      <AQ2>
31080 LOOP                                             <6B9>
31085 NA$(IL,0)=YD$: IL=IL+1                           <23J>
31090 IK=IK+1: IF IK=8 THEN EXIT                       <CMR>
31095 LOOP                                             <B9H>
```

Listing 1. »CH-VERWALTUNG« (Fortsetzung)

```
31100 IF CR%=1 THEN EXIT                                <1N2>
31105 PRINT#15,"B-P 2 0"                                <OTT>
31110 GET #2,XD$: GET #2,YD$: IF XD$="" THEN EXIT
                                                        <OP2>
31115 IJ=ASC(YD$)                                       <S44>
31120 LOOP                                              <6C9>
31125 IF DS>0 THEN 31175                                <GQM>
31130 CLOSE 2: CLOSE 15: IL=IL-1: IJ=1: IK=1            <TT6>
31135 IF IL=0 THEN 31185                                <NMP>
31140 DO WHILE IJ<=IL                                   <9T4>
31145 YD$="K"+NA$(IJ,0)                                 <TI5>
31150 DOPEN #2,(YD$): IF DS>0 THEN CR%=1: EXIT          <QUR>
31155 INPUT#2,YD$: IF NA$(IJ,0)<>YD$ THEN IJ=IJ+1
      : GOTO 31165                                      <6VH>
31160 INPUT#2,NA$(IJ,1): NA$(IK,0)=NA$(IJ,0): NA$
      (IK,1)=NA$(IJ,1): IK=IK+1: IJ=IJ+1                <JAE>
31165 DCLOSE #2: LOOP                                   <B1O>
31170 IF CR%=0 THEN 31180                               <6EJ>
31175 GOSUB 4001: IF CR%=0 THEN 31001: ELSE RETUR
      N                                                 <70I>
31180 IL=IK-1                                           <3CF>
31185 IF CX%=0 THEN M1$=" / INHALTSVERZEICHNIS DE
      R DATEIDISKETTE": GOSUB 10011                     <LPE>
31190 CHAR ,0,3,CL$                                     <AH0>
31195 CHAR 1,0,22,CL$+"++++++++++++++++++++++++++++
      ++++++++++++++++++++++++++++++++++++++++++
      +++++++++++"                                      <G91>
31200 IF CX%=0 THEN PRINT " (RVSON) D (RVOFF) - I
      NHALTSVERZEICHNIS AUF DRUCKER"                    <HSN>
31201 PRINT " (RVSON) + (RVOFF) / (RVSON) - (RVOF
      F) - BLÄTTERN(2HOME)"                             <CQA>
31205 WINDOW 0,4,79,21,1                                <8OS>
31210 IK=1: IJ=1                                        <9PN>
31215 IF IL=0 THEN BEGIN                                <9L3>
31220 PRINT " KEINE TURNIERDATEI AUF DIESER DISKE
      TTE ABGESPEICHERT!"                               <CSR>
31225 PRINT "(2HOME)": GET KEY XT$                      <FG3>
31230 BEND : RETURN                                     <8EF>
31235 DO WHILE IJ<=IL                                   <LT4>
31240 IM=IJ: GOSUB 31280: PRINT                         <V69>
31245 IF PEEK(235)=21 THEN EXIT                         <DI3>
31250 IJ=IJ+1: LOOP                                     <57P>
31255 PRINT "(2HOME)": GET KEY XT$: IF INSTR("-+D
      ",XT$)=0 THEN RETURN                              <0MT>
31260 IF XT$="D" AND CX%=0 THEN GOSUB 38001             <4S6>
31265 IF XT$="+" AND IJ<IL THEN IJ=IK+1: IK=IK+1:
       IM=IJ: WINDOW 0,4,79,21: CHAR ,0,17,CR$: G
      OSUB 31280: PRINT : GOTO 31255                    <QA9>
31270 IF XT$="-" AND IK>1 THEN IK=IK-1: IJ=IJ-1:
      IM=IK: WINDOW 0,4,79,21: PRINT CHR$(27)"I";
      : GOSUB 31280: GOTO 31255                         <MJH>
31275 GOTO 31255                                        <CD5>
31280 PRINT "(2SPACE)" RIGHT$(STR$(IM),3);TAB(5)N
      A$(IM,0);TAB(18)"-"NA$(IM,1);: BANK 15: SYS
       2848: RETURN                                     <5R6>
32001 CHAR ,0,23,CL$: PRINT " GEBEN SIE NAMEN UND
       ID DER ZU FORMATIERENDEN DISKETTE EIN!"          <0A9>
32002 PRINT " (RVSON) NAME: (RVOFF)";TAB(30)"(RVS
      ON) ID:(SHF.SPACE,RVOFF)";: XD$="": YD$=""        <5AJ>
32003 EY%=24: EX%=9: EL%=16: E$(2)="": E$(0)=YD$:
       E$(3)="ABCDEFGHIJKLMNOPQRSTUVWXYZ "+ZA$          <PVG>
32004 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN           <5PA>
32005 YD$=E$(1): IF YD$="" THEN GOSUB 2006: GOTO
      32003                                             <B6H>
32006 EX%=37: EL%=2: E$(0)=XD$: E$(2)="(SYNTH.:14
      5)"                                               <IF1>
32007 GOSUB 3001: IF XT$="(SYNTH.:145)" THEN 3200
      3                                                 <BRD>
32008 IF XT$=CHR$(27) THEN RETURN                       <IDO>
32009 XD$=E$(1)                                         <7BV>
32010 CHAR ,0,23,CL$: PRINT " VORSICHT - DIE AUF
      DER EINGELEGTEN DISKETTE ABGESPEICHERTEN DA
      TEN GEHEN"                                        <FT2>
32011 PRINT " VERLOREN!(2SPACE)*(2SPACE)WEITER MI
      T (RVSON) RTN (RVOFF)";: GET KEY XT$: IF XT
      $<>CHR$(13) THEN RETURN                           <BSB>
32012 OPEN 15,8,15                                      <C0B>
32013 PRINT#15,"N:"+YD$+","+XD$                         <0DM>
32014 IF DS=0 THEN CLOSE 15: RETURN                     <ED5>
32015 GOSUB 4001: IF CR%=0 THEN 32012: ELSE RETUR
      N                                                 <UBI>
33001 M1$=" / TURNIERDATEI LÖSCHEN": GOSUB 10011        <KOC>
33002 CX%=1: GOSUB 31001: CX%=0                         <JRR>
33005 IF CR%=1 OR IL=0 THEN RETURN                      <B57>
33015 CHAR ,0,23,CL$                                    <TBG>
33020 PRINT " GEBEN SIE DIE LFD. NUMMER DER ZU LÖ
      SCHENDEN DATEI EIN!"                              <UD4>
33025 EY%=24: EX%=1: EL%=3: E$(0)="": E$(2)="": E
      $(3)=ZA$                                          <KJT>
33030 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN           <0PU>
33035 IM=VAL(E$(1)): IF IM=0 OR IM>IL THEN GOSUB
      2006: GOTO 33030                                  <J9R>
33040 CHAR ,0,23,CR$: GOSUB 31280: PRINT                <N20>
33045 PRINT " LÖSCHEN? "JN$;                            <BCN>
33050 GET KEY XT$: IF XT$<>"J" THEN 33015               <CQS>
33055 YD$="K"+NA$(IM,0): SCRATCH(YD$): IF DS>2 TH
      EN 33095                                          <0M2>
33060 YD$="S"+NA$(IM,0): SCRATCH(YD$): IF DS>2 TH
      EN 33095                                          <RM2>
33065 YD$="T"+NA$(IM,0): SCRATCH(YD$): IF DS<2 TH
      EN 33001                                          <GM9>
33095 GOSUB 4001: IF CR%=0 THEN 33055: ELSE : GOT
      O 33001                                           <VVR>
34001 M1$=" / TURNIERDATEI UMBNENNEN": GOSUB 1001
      1                                                 <BML>
34010 CX%=1: GOSUB 31001: CX%=0: IF CR%=1 OR IL=0
       THEN RETURN                                      <9EK>
34015 CHAR ,0,23,CL$                                    <TNJ>
34020 PRINT " GEBEN SIE DIE LFD. NUMMER DER UMZUB
      ENNENDEN DATEI EIN!"                              <4I7>
34025 EY%=24: EX%=1: EL%=3: E$(0)="": E$(2)="": E
      $(3)=ZA$                                          <LBQ>
34030 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN           <SQE>
34035 IM=VAL(E$(1)): IF IM=0 OR IM>IL THEN GOSUB
      2006: GOTO 33030                                  <FAB>
34040 CHAR ,0,23,CR$: GOSUB 31280: PRINT                <MQ7>
34045 PRINT " UMBENENNEN?(SHF.SPACE)"JN$;               <MSU>
34050 GET KEY XT$: IF XT$<>"J" THEN 34015               <B63>
34055 CHAR ,0,24,CL$                                    <JIB>
```

Listing 1. »CH-VERWALTUNG« (Fortsetzung)


Bildnachweis: CDZ-Film, Berlin
GEOS™
GRAPHIC ENVIRONMENT OPERATING SYSTEM
DER NEUE BETRIEBSSYSTEM-STANDARD FÜR
DIE COMMODORE COMPUTER 64, 64c UND 128.
deskTop: Organisations- und Dateiverwaltungsprogramm
geoWrite: einfach zu handhabendes, auf Grafiken basierendes Textverarbeitungssystem
geoPaint: leistungsstarker Grafikeditor
diskTurbo: schnelles Dateiladeprogramm
Hilfsprogramme: Wecker, Taschenrechner, Notizblock, Foto-Album, Text-Album
GEOS
GEOWRITE WORKSHOP
GEOFILE
DESKPACK
Eine neue Welt
für den C64/C128

```
34060 PRINT " {RVSON} NEUER NAME: {RVOFF}";                <F3H>
34065 EX%=15: EL%=13: E$(0)=NA$(IM,0): E$(2)="":
      E$(3)=".ABCDEFGHIJKLMNOPQRSTUVWXYZ "+ZA$             <DD6>
34070 GOSUB 3001: IF XT$=CHR$(27) THEN 34015              <SVF>
34075 IF E$(1)="" THEN GOSUB 2006: GOTO 34070             <SF5>
34080 TU%=0: TU$="": DR%=0: YD$="A"+NA$(IM,0)             <G0J>
34081 DOPEN #2,(YD$)                                      <F7U>
34082 INPUT#2,XX$,XY$,XX%,XY%: IF DS>0 THEN 34135         <4H2>
34083 DCLOSE #2                                           <CL5>
34084 SCRATCH(YD$)                                        <R3D>
34085 XX$=E$(1): YD$="A"+XX$                              <BUV>
34086 DOPEN #2,(YD$),W                                    <D7V>
34087 PRINT#2,XX$;CHR$(13);XY$;CHR$(13);XX%;CHR$(
      13);XY%: IF DS>0 THEN 34135                         <9P4>
34088 DCLOSE #2                                           <DP5>
34090 YD$="S"+NA$(IM,0): XD$="S"+E$(1)                    <5P7>
34095 RENAME(YD$) TO(XD$): IF DS>0 AND DS<>62 THE
      N 34135                                             <LA1>
34100 YD$="T"+NA$(IM,0): XD$="T"+E$(1)                    <L32>
34105 RENAME(YD$) TO(XD$): IF DS>0 AND DS<>62 THE
      N 34135                                             <3A1>
34130 GOTO 34001                                          <3TI>
34135 GOSUB 4001: IF CR%=0 THEN 34080: ELSE GOTO
      34001                                               <L0V>
35001 GOSUB 5011: IF CR%=1 THEN RETURN                    <EFQ>
35013 GOSUB 5001: IF CR%=1 THEN RETURN                    <FFU>
35019 OPEN 5,5: OPEN 4,4                                  <J11>
35020 PRINT#5,CHR$(27);CHR$(108);CHR$(7);                 <830>
35025 PP$="> LIST OF PLAYERS <": GOSUB 35900              <T0Q>
35029 GOSUB 35910                                         <BMJ>
35030 IX=1                                                <K21>
35035 DO WHILE IX<=SR%(0): BANK 0: II=SR%(IX)             <JRQ>
35040 PRINT#4,RIGHT$("{2SPACE}"+STR$(II),3)"{CTRL
      +I}";                                               <0LF>
35045 PRINT#4,NA$(II,0)"{CTRL+I}"NA$(II,1);: IF N
      A$(II,2)<>"" THEN PRINT#4,", "NA$(II,2);            <IIM>
35050 PRINT#4,"{CTRL+I}"NA$(II,3)"{CTRL+I}";              <F80>
35055 IF NA%(II,0)>0 THEN PRINT#4,RIGHT$("{2SPACE
      }"+STR$(NA%(II,0)),3);                              <B6U>
35060 PRINT#4,"{CTRL+I}";: IF NA%(II,1)>0 THEN PR
      INT#4,RIGHT$("{3SPACE}"+STR$(NA%(II,1)),4);         <5GF>
35065 PRINT#4: PP%=PP%+1: IF PP%=60 THEN PRINT#5,
      CHR$(12);: GOSUB 35900: GOSUB 35910                 <IVJ>
35070 IX=IX+1: LOOP : PRINT#5,CHR$(12): CLOSE 4:
      CLOSE 5                                             <7P1>
35075 RETURN                                              <BF7>
35100 CHAR ,0,23,CL$: CR%=0                               <3II>
35105 PRINT " SORTIERKRITERIUM:                           <P9T>
35110 PRINT " {RVSON} N {RVOFF} - NAME{2SPACE}*{2
      SPACE,RVSON} V {RVOFF} - VEREIN{2SPACE}*{2S
      PACE,RVSON} Z {RVOFF} - WERTUNGSZAHLEN";            <9MI>
35115 GET KEY XT$: IF XT$=CHR$(27) THEN CR%=1: RE
      TURN                                                <D4V>
35120 YT%=INSTR("NVZ",XT$)                                <KMI>
35125 CHAR ,0,23,CL$+" SPIELERLISTE WIRD SORTIERT
      !"                                                  <K65>
35130 BANK 0: POKE 5120,MS%: SYS 5414                     <CNL>
35135 ON YT% GOSUB 35150,35150,35180                      <3SQ>
35140 FOR IX=1 TO MS%: SR%(IX)=PEEK(5120+IX): NEX
      T IX: SR%(0)=MS%: RETURN                            <J2H>
35150 FOR IX=1 TO MS%: ON YT% GOSUB 35165,35170           <LFV>
35155 FOR IY=1 TO LEN(XB$): POKE 53248+(IX-1)*37+
      IY-1,ASC(MID$(XB$,IY,1)): NEXT IY                   <61U>
35160 NEXT IX: SYS 5392: RETURN                           <QRK>
35165 XB$=NA$(IX,1)+CHR$(0)+NA$(IX,2)+CHR$(13): R
      ETURN                                               <EAT>
35170 XB$=NA$(IX,3)+CHR$(13): RETURN                      <IPN>
35180 FOR IX=1 TO MS%: FOR IY=0 TO 1                      <L7C>
35185 XX%=NA%(IX,IY): IF IY=0 AND XX%=0 THEN XX%=
      999                                                 <029>
35186 IF IY=1 AND XX%=0 AND NA%(IX,0)<130 THEN XX
      %=SGN(NA%(IX,0))*(2839-8*NA%(IX,0))                 <445>
35190 XY%=INT(XX%/256): XZ%=XX%-XY%*256                   <NL3>
35195 POKE 52992-IY*512+IX,XZ%: POKE 52736-IY*512
      +IX,XY%                                             <JKE>
35200 NEXT IY,IX: SYS 5403                                <LM6>
35205 RETURN                                              <D97>
35900 PRINT#5,CHR$(27)"!" CHR$(33+PM%);                   <KBV>
35901 XT%=INSTR(TU$," ",30): IF XT%=0 THEN XT%=LE
      N(TU$)+1                                            <J60>
35902 XX$=LEFT$(TU$,XT%-1): XY$=MID$(TU$,XT%+1,60
      )                                                   <5CT>
35903 PRINT#4,XX$: PRINT#4,XY$: PRINT#5,CHR$(27)"
      J" CHR$(18);: PRINT#4,PP$                           <ES8>
35904 PRINT#5,CHR$(27)"-1";                               <71O>
35905 FOR IL=1 TO 43: PRINT#4," ";: NEXT IL: PRIN
      T#5,CHR$(27)"-0";                                   <3DL>
35906 PRINT#4: PRINT#4: RETURN                            <176>
35910 PRINT#5,CHR$(27)"!{CTRL+A}" CHR$(27)"D{WHIT
      E,CTRL+I}-FK" CHR$(0);                              <CC1>
35911 PRINT#5,CHR$(27)"!" CHR$(4+PM%);                    <FF8>
35912 PRINT#4,"NR.{CTRL+I}{TITLE} NAME, FORENAME{
      CTRL+I}CLUB/TOWN/NATION{CTRL+I}INGO{CTRL+I}
      ELO"                                                <BT9>
35913 PRINT#4                                             <1G1>
35914 PRINT#5,CHR$(27)"!" CHR$(1+PM%);: PP%=10: R
      ETURN                                               <KHL>
36001 GOSUB 5021: ON XT% GOTO 36002,18001: RETURN
                                                          <5OE>
36002 GOSUB 5031: IF CR%=1 THEN RETURN                    <NFI>
36025 PP%=XT%-1                                           <D0T>
36026 GOSUB 5011: IF CR%=1 THEN RETURN                    <LFQ>
36027 GOSUB 5001: IF CR%=1 THEN RETURN                    <LFU>
36045 CR%=PP%: OPEN 5,5: OPEN 4,4                         <DSH>
36050 PRINT#5,CHR$(27);CHR$(108);CHR$(7);                 <U10>
36055 PP$="> PAIRINGS & RESULTS / ROUND"+STR$(DR%
      )+" <"                                              <66L>
36060 GOSUB 35900: GOSUB 36900                            <MA4>
36065 IX=1: BANK 1                                        <BCV>
36070 DO WHILE IX<=SR%(0)                                 <U22>
36075 PN%=SR%(IX): GOSUB 20101                            <AQK>
36080 GOSUB 36800                                         <9US>
36085 XX$=LEFT$(XX$+SP$,39)+" ": XX$=LEFT$(XX$+XY
      $+SP$,76)+" "+XZ$                                   <U0G>
36090 PRINT#4,XX$                                         <5MI>
36130 PP%=PP%+1: IF PP%=60 THEN PRINT#5,"{CTRL+L}
      ";: GOSUB 35900: GOSUB 36900                        <R6F>
36135 IX=IX+1: LOOP                                       <NFO>
36140 PRINT#5,"{CTRL+L}";                                 <5RE>
36145 IF CR%=1 THEN CR%=0: GOTO 36060: ELSE CLOSE
       4: CLOSE 5: RETURN                                 <LAA>
36800 XX$=RIGHT$("{2SPACE}"+STR$(PN%),3)+"{2SPACE
      }": XY$=" - "                                       <PFJ>
36805 IF CR%=0 THEN 36820                                 <5IS>
36810 XX$=XX$+"("+RIGHT$("{2SPACE}"+STR$(SP%),3)+
      ") "                                                <3PN>
36815 IF EG%<8 THEN XY$=XY$+"("+RIGHT$("{2SPACE}"
      +STR$(SO%),3)+"){SHF.SPACE}"                        <A89>
36820 IF EG%=8 THEN XY$=XY$+"SPIELFREI"                   <N10>
36825 IF EG%=9 THEN XY$=XY$+"ZURUCKGEZOGEN"               <C6L>
36830 XX$=XX$+NA$(SP%,1): IF NA$(SP%,2)<>"" THEN
      XX$=XX$+", "+NA$(SP%,2)                             <SJD>
36835 IF EG%<8 THEN XY$=XY$+NA$(SO%,1): IF NA$(SO
      %,2)<>"" THEN XY$=XY$+", "+NA$(SO%,2)               <VT3>
36845 XZ$=EG$(EG%)                                        <1GR>
36850 RETURN                                              <1M7>
36900 PRINT#5,CHR$(27)"!" CHR$(1+PM%);                    <MUE>
36905 PP%=8: RETURN                                       <2QN>
37001 GOSUB 5021: ON XT% GOTO 37004,17001: RETURN
                                                          <Q8E>
37004 GOSUB 5031: IF CR%=1 THEN RETURN                    <PFI>
37005 PP%=XT%-1: CS%=0                                    <5FI>
37008 GOSUB 5011: IF CR%=1 THEN RETURN                    <NFQ>
37015 GOSUB 5001: IF CR%=1 THEN RETURN                    <PFU>
37030 CR%=PP%: OPEN 5,5: OPEN 4,4                         <TQH>
37035 PRINT#5,CHR$(27);CHR$(108);CHR$(7);                 <V14>
37040 PP$="> RANKING LIST AFTER THE"+STR$(DR%)+".
      ROUND <": IF DR%=MR% AND MO%=4 THEN PP$=">
      FINAL RANKING LIST <"                               <8NB>
37045 GOSUB 35900: GOSUB 37900                            <JEG>
37050 II=1: P1=0: H1=0: B1=0: A1=0: BANK 1                <P5C>
37055 DO WHILE II<=MS%                                    <77Q>
37056 GOSUB 37700: GOSUB 37800                            <VQ2>
37120 PRINT#4,XX$                                         <VMI>
37130 II=II+1: PP%=PP%+1                                  <4J4>
37135 IF PP%=60 THEN PRINT#5,"{CTRL+L}";: GOSUB 3
      5900: GOSUB 37900                                   <F7A>
37140 LOOP                                                <U39>
37145 PRINT#5,"{CTRL+L}";: IF CR%=1 THEN CR%=0: G
      OTO 37045: ELSE CLOSE 4: CLOSE 5: RETURN            <BPF>
37700 SO%=PEEK(DEC("3301")+II): SP%=PEEK(DEC("330
      0")+II)                                             <JJO>
37705 P0=P1: H0=H1: B0=B1: A0=A1                          <1BH>
37710 P1=PEEK(DEC("3400")+SP%)/2                          <TUG>
37715 P2=PEEK(DEC("3400")+SO%)/2                          <15B>
37720 H1=PEEK(DEC("3500")+SP%)                            <T3P>
37725 H2=PEEK(DEC("3500")+SO%)                            <A57>
37730 B1=(PEEK(DEC("3600")+SP%)+PEEK(DEC("0600")+
      SP%)*256)/2                                         <MMN>
37735 B2=(PEEK(DEC("3600")+SO%)+PEEK(DEC("0600")+
      SO%)*256)/2                                         <EN8>
37740 A1=(PEEK(DEC("3700")+SP%)+PEEK(DEC("0700")+
      SP%)*256)/2                                         <2S7>
37745 A2=(PEEK(DEC("3700")+SO%)+PEEK(DEC("0700")+
      SO%)*256)/2                                         <QTO>
37750 RETURN                                              <027>
37800 XX$="{6SPACE}": IF(PP%=10 AND CS%=0) OR P1<
      >P0 OR H1<>H0 OR B1<>B0 OR A1<>A0 THEN XX$=
      RIGHT$("{3SPACE}"+STR$(II)+".",4)+"{2SPACE}
      "                                                   <BON>
37805 IF CR%=1 THEN XX$=XX$+"("+RIGHT$("{2SPACE}"
      +STR$(SP%),3)+") "                                  <FSH>
37810 XX$=XX$+LEFT$(NA$(SP%,0)+"{3SPACE}",3)+" "          <HIH>
37815 XX$=XX$+NA$(SP%,1): IF NA$(SP%,2)<>"" THEN
      XX$=XX$+", "+NA$(SP%,2)                             <KRC>
37820 XX$=LEFT$(XX$+SP$,49)+" "                           <3F8>
37825 XY=P1: GOSUB 37891                                  <RQL>
37830 XX$=XX$+RIGHT$("{4SPACE}"+XY$,3)+LEFT$(XZ$+
      "{2SPACE}",2)                                       <VRN>
37835 IF H1>0 THEN XX$=XX$+"+"+RIGHT$(" "+STR$(H1
      ),2)+"H"                                            <1PO>
37840 XX$=LEFT$(XX$+SP$,61)                               <2CN>
37841 IF B1=0 OR DR%=1 OR(P0<>P1 OR H0<>H1) AND(P
      1<>P2 OR H1<>H2) THEN RETURN                        <C04>
37850 XY=B1: GOSUB 37891                                  <UIG>
37855 XX$=XX$+RIGHT$("{4SPACE}"+XY$,4)+XZ$                <436>
37860 XX$=LEFT$(XX$+SP$,68)                               <G6U>
37865 IF A1=0 OR DR%=2 OR(P0<>P1 OR H0<>H1 OR B0<
      >B1) AND(P1<>P2 OR H1<>H2 OR B1<>B2) THEN R
      ETURN                                               <TI5>
37870 XY=A1: GOSUB 37891                                  <Q21>
37875 XX$=XX$+RIGHT$("{4SPACE}"+XY$,4)+XZ$                <4J4>
37890 RETURN                                              <167>
37891 XY$=LEFT$(STR$(XY),INSTR(STR$(XY)+".",".")-
      1): XZ$=MID$(STR$(XY),INSTR(STR$(XY)+".",".
      "),10): RETURN                                      <4O7>
37900 PRINT#5,CHR$(27)"!{CTRL+A}" CHR$(27)"D{CTRL
      +F}2E" CHR$(0);                                     <03E>
37905 PRINT#5,CHR$(27)"!" CHR$(4+PM%);                    <E3D>
37910 PRINT#4,"RANK{CTRL+I}PLAYER{CTRL+I}POINTS{C
      TRL+I}BHZ{CTRL+I}MBHZ"                              <EDH>
37915 PRINT#4                                             <0S1>
37920 PRINT#5,CHR$(27)"!" CHR$(1+PM%);: PP%=10: R
      ETURN                                               <SPK>
38001 GOSUB 5001: IF CR%=1 THEN RETURN                    <73V>
38020 PM%=0: OPEN 4,4: OPEN 5,5                           <F1M>
38030 PRINT#5,CHR$(27);CHR$(108);"{CTRL+J}";              <GFD>
38040 PRINT#5,CHR$(27)"!!";                               <218>
38045 PRINT#4,"INHALTSVERZEICHNIS DER DATEIDISKET
      TE": PRINT#4                                        <CMC>
38050 PRINT#5,CHR$(27)"!{CTRL+A}";                        <Q28>
38051 PRINT#4,"NAME: "NA$(0,0)                            <GKV>
38052 PRINT#4,"ID:{3SPACE}"NA$(0,1)                       <D4K>
38053 PRINT#4: PRINT#4                                    <L2N>
38055 II=1                                                <UBB>
38060 DO WHILE II<=IL                                     <EPO>
38070 PRINT#4,RIGHT$("{2SPACE}"+STR$(II),3)+"{2SP
      ACE}"+LEFT$(NA$(II,0)+"{13SPACE}",13)+" - "
      +NA$(II,1)                                          <0J1>
38080 II=II+1: LOOP                                       <05R>
38090 PRINT#5,"{CTRL+L}";: CLOSE 4: CLOSE 5: RETU
      RN                                                  <5C3>
39001 CHAR ,0,23,CL$: PRINT " AUSGABE VON SPIELER
       NR.:{5SPACE}BIS NR.:"                              <JAJ>
39002 EX%=26: EY%=23: EL%=3: E$(0)="": E$(2)="":
      E$(3)=ZA$                                           <EU2>
39003 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN             <ACK>
39004 IX=VAL(E$(1)): IF IX=0 OR IX>MS% THEN GOSUB
       2006: GOTO 39002                                   <BSQ>
39005 EX%=39: E$(2)="{SYNTH.:145}"                        <9JN>
39006 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN             <BSU>
39007 IF XT$="{SYNTH.:145}" THEN 39001                    <JUU>
39008 IY=VAL(E$(1)): IF IX=0 OR IY>MS% OR IY<IX T
      HEN GOSUB 2006: GOTO 39006                          <P9S>
39009 GOSUB 5011: IF CR%=1 THEN RETURN                    <63R>
39010 GOSUB 5001: IF CR%=1 THEN RETURN                    <C3V>
39011 OPEN 5,5: OPEN 4,4                                  <LHR>
39012 PRINT#5,CHR$(27);CHR$(108);CHR$(7);                 <CF8>
```

```
39013 PP$="> CH-SYSTEM-KARTE <"                                  <5AJ>
39014 DO WHILE IX<=IY                                             <69Q>
39015 GET XT$: IF XT$=CHR$(27) THEN EXIT                          <VMD>
39016 SP%=IX                                                      <SDR>
39023 GOSUB 35900: GOSUB 35910                                    <ORP>
39024 PRINT#4,RIGHT$(" "+STR$(SP%),3)"(CTRL+I)"NA
      $(SP%,0)"(CTRL+I)"NA$(SP%,1);: IF NA$(SP%,2
      )<>"" THEN PRINT#4,", "NA$(SP%,2);                          <HNF>
39025 PRINT#4,"(CTRL+I)"NA$(SP%,3)"(CTRL+I)";: IF
       NA%(SP%,0)>0 THEN PRINT#4,USING "###";NA%(
      SP%,0);                                                     <91N>
39026 PRINT#4,"(CTRL+I)";: IF NA%(SP%,1)>0 THEN P
      RINT#4,USING "####";NA%(SP%,1);                             <2VV>
39027 PRINT#4: PRINT#5,CHR$(27)"-1";: PRINT#4,LEF
      T$(SP$+SP$,87): PRINT#5,CHR$(27)"-0";: PRIN
      T#4                                                         <SKT>
39028 PRINT#5,CHR$(27)"!(CTRL+A)" CHR$(27)"D(CTRL
      +D)04:=H" CHR$(0);CHR$(27)"!" CHR$(4+PM%);                  <B6J>
39029 PRINT#4,"RD.(CTRL+I)GEGNER(CTRL+I)(INGO/ELO
      )(CTRL+I)FA.(CTRL+I)ERGEBNIS(CTRL+I)PUNKTE"                 <7I0>
39030 PRINT#5,CHR$(27)"!" CHR$(1+PM%);: PRINT#4                   <3IK>
39031 IJ=IX: GOSUB 23026                                          <3VP>
39032 FOR IL=1 TO RD%                                             <I2C>
39033 GOSUB 23090                                                 <U2P>
39034 PRINT#4,RIGHT$(STR$(IL),2)"(CTRL+I)";                       <KJ3>
39035 IF SO%=0 THEN XX$="ZURÜCKGEZOGEN": GOTO 390
      40                                                          <NTF>
39036 IF SO%=255 THEN XX$="SPIELFREI": GOTO 39040                 <TOP>
39037 IF SO%=256 THEN XX$="NOCH NICHT GEPAART": G
      OTO 39040                                                   <86P>
39038 XX$=RIGHT$(" "+STR$(SO%),3)+"(2SPACE)"+LEFT
      $(NA$(SO%,0)+"(4SPACE)",4)+NA$(SO%,1): IF N
      A$(SO%,2)<>"" THEN XX$=XX$+", "+NA$(SO%,2)                  <CJK>
39039 XX$=LEFT$(XX$+SP$,38)+" "+LEFT$(NA$(SO%,3),
      3)                                                          <QLA>
39040 PRINT#4,XX$;: IF SO%=0 OR SO%>254 THEN PRIN
      T#4,"(CTRL+I) (CTRL+I)";: GOTO 39043                        <KD8>
39041 PRINT#4,"(CTRL+I)";: IF NA%(SO%,0)>0 THEN P
      RINT#4,USING "###";NA%(SO%,0);                              <L74>
39042 PRINT#4,"(CTRL+I)";: IF NA%(SO%,1)>0 THEN P
      RINT#4,USING "####";NA%(SO%,1);                             <P0E>
39043 PRINT#4,"(CTRL+I)"FA$(FA%)"(CTRL+I)"EG$(EG%
      )"(CTRL+I)";: PRINT#4,USING "##.#";P0;                      <BA0>
39044 IF H0>0 THEN PRINT#4," +" RIGHT$(STR$(H0),2
      )"H";                                                       <HUM>
39045 PRINT#4: NEXT IL                                            <9B5>
39046 PRINT#5,"(CTRL+L)";: IX=IX+1: LOOP                          <F2K>
39047 CLOSE 4: CLOSE 5: RETURN                                    <OUQ>
40001 CX%=0: M0$="SPIELERVERWALTUNG": M1$="": DR%
      =0: GOSUB 10005                                             <0AT>
40002 CHAR ,0,3,CL$                                               <I93>
40005 PRINT "(RVSON) NR.T(TITEL) NAME, VORNAME(18
      SPACE)TVEREIN (NATION/STADT)(3SPACE)TINGOTE
      LO(2SPACE)"                                                 <A65>
40010 CHAR 1,0,22,"T+++T+++T+++++++++++++++++++++
      +++++++++++++T++++++++++++++++++++++++++T+++
      +T+++++U"                                                   <FET>
40015 MM%=5: IF MS%>0 THEN GOSUB 40134                            <HD6>
40020 GOTO 41001                                                  <TCU>
40101 IF SP%>0 THEN PRINT " " RIGHT$("(2SPACE)"+S
      TR$(SP%),3);                                                <ML1>
40102 PRINT TAB(5)NA$(SP%,0);TAB(9)NA$(SP%,1);                    <RAS>
40105 IF NA$(SP%,2)<>"" THEN PRINT ", "NA$(SP%,2)
      ;                                                           <DQC>
40110 PRINT TAB(45)NA$(SP%,3);                                    <BPH>
40115 IF NA%(SP%,0)>0 THEN PRINT TAB(70) RIGHT$("
      (2SPACE)"+STR$(NA%(SP%,0)),3);                              <QK2>
40120 IF NA%(SP%,1)>0 THEN PRINT TAB(75) RIGHT$("
      (3SPACE)"+STR$(NA%(SP%,1)),4);                              <JC9>
40121 IF CA%=1 THEN CA%=0: RETURN                                 <BDF>
40125 BANK 15: SYS 2848: RETURN                                   <6Q0>
40130 CHAR ,0,23,CL$: PRINT " SOLL LISTE SORTIERT
       AUSGEGEBEN WERDEN? "JN$: GET KEY XT$: IF X
      T$=CHR$(27) THEN RETURN                                     <LTM>
40131 IF XT$<>"J" THEN 40134                                      <VIC>
40132 GOSUB 35100: IF CR%=1 THEN RETURN                           <S4F>
40133 GOTO 40135                                                  <TSP>
40134 SR%(0)=MS%: FOR IJ=1 TO MS%: SR%(IJ)=IJ: NE
      XT IJ                                                       <0V1>
40135 M1$=" / SPIELERLISTE": GOSUB 10011                          <7EL>
40140 IJ=1                                                        <PNL>
40145 WINDOW 0,5,79,20,1: IK=IJ                                   <9FT>
40150 DO WHILE IJ<=SR%(0)                                         <C82>
40155 SP%=SR%(IJ): GOSUB 40101: PRINT                             <JH7>
40160 IF IJ=IK+14 THEN EXIT                                       <S56>
40165 IJ=IJ+1: LOOP : IJ=IK+14                                    <4F3>
40170 MM%=PEEK(235): PRINT "(2HOME)"                              <NPI>
40175 CHAR ,0,23,CL$                                              <SRM>
40180 PRINT " (RVSON) D (RVOFF) - LISTE AUF DRUCK
      ER": PRINT BX$;                                             <6N3>
40185 PRINT "(2HOME)": CHAR 1,0,21,CR$: GET KEY X
      T$                                                          <F6V>
40190 IF XT$="D" THEN GOSUB 35001: GOTO 40175                     <9TC>
40191 IF XT$="P" AND IJ<SR%(0) THEN IJ=IJ+1: GOTO
       40145                                                      <RV4>
40192 IF XT$="P" AND IK-15>0 THEN IJ=IK-15: GOTO
      40145                                                       <GFM>
40193 IF XT$="(HOME)" THEN IJ=1: GOTO 40145                       <7TJ>
40194 IF XT$="(CLR)" AND SR%(0)>14 THEN IJ=SR%(0)
      -14: GOTO 40145                                             <9M6>
40195 IF XT$="+" AND IJ<SR%(0) THEN IJ=IJ+1: IK=I
      K+1: SP%=SR%(IJ): WINDOW 0,5,79,20: CHAR ,0
      ,15,CR$: GOSUB 40101: PRINT : GOTO 40185                    <NSN>
40200 IF XT$="-" AND IK>1 THEN IK=IK-1: IJ=IJ-1:
      SP%=SR%(IK): WINDOW 0,5,79,20: CHAR ,0,0,CH
      R$(27)+"I": GOSUB 40101: GOTO 40185                         <LV2>
40202 IF INSTR("+-PP",XT$)>0 THEN 40185                           <2D5>
40205 RETURN                                                      <QH7>
40220 M1$=" / SPIELER SUCHEN": GOSUB 10011                        <FI7>
40222 NA$(0,0)="": NA$(0,1)="": NA$(0,2)="": NA$(
      0,3)="": NA%(0,0)=0: NA%(0,1)=0                             <FA6>
40225 CHAR ,0,23,CL$: PRINT " GEBEN SIE DIE DATEN
       DES GESUCHTEN SPIELERS EIN!"                               <KPR>
40230 GOSUB 40295: IF CR%=1 THEN RETURN                           <1G9>
40235 CHAR ,0,23,CL$: XB$=NA$(0,0)+NA$(0,1)+NA$(0
      ,2)+NA$(0,3)                                                <5D3>
40240 IJ=1: SP%=1                                                 <DJJ>
40245 DO WHILE SP%<=MS%                                           <365>
40250 YB$="": FOR IK=0 TO 3: YB$=YB$+LEFT$(NA$(SP
      %,IK),LEN(NA$(0,IK))): NEXT IK                              <GML>
40255 IF XB$=YB$ THEN SR%(IJ)=SP%: IJ=IJ+1                        <SDD>
40275 SP%=SP%+1: LOOP                                             <BJ4>
40276 IF IJ=1 THEN RETURN                                         <2LJ>
40277 SR%(0)=IJ-1: GOTO 40140                                     <UHF>
40280 RETURN                                                      <I47>
40295 CHAR 1,0,21,CR$: SP%=0: GOSUB 40101                         <HEA>
40300 EY%=21: CR%=1                                               <MPC>
40305 EX%=5: EL%=3: E$(0)=NA$(0,0): E$(2)="": E$(
      3)="HFIGM"                                                  <40S>
40310 GOSUB 3001                                                  <1B6>
40315 IF XT$=CHR$(27) THEN RETURN                                 <U10>
40320 NA$(0,0)=E$(1)                                              <TNG>
40325 EX%=9: EL%=35: E$(2)="(SYNTH.:145)": E$(3)=
      BS$+ZA$+","                                                 <J0L>
40330 E$(0)=NA$(0,1): IF NA$(0,2)<>"" THEN E$(0)=
      E$(0)+", "+NA$(0,2)                                         <A66>
40335 GOSUB 3001                                                  <136>
40340 IF XT$=CHR$(27) THEN RETURN                                 <VTO>
40345 IF XT$="(SYNTH.:145)" THEN 40300                            <A9D>
40350 XT%=INSTR(E$(1)+",",",")                                    <VQA>
40355 NA$(0,1)=LEFT$(E$(1),XT%-1)                                 <2V3>
40360 NA$(0,2)=MID$(E$(1),XT%+1,35)                               <9D0>
40365 IF NA$(0,2)="" THEN 40380                                   <PF7>
40370 IF ASC(NA$(0,2))<>32 OR INSTR(NA$(0,2),",")
      >0 THEN GOSUB 2006: GOTO 40330                              <U4U>
40375 NA$(0,2)=MID$(NA$(0,2),2,35)                                <JJJ>
40380 EX%=45: EL%=24: E$(3)=BS$+ZA$: E$(0)=NA$(0,
      3): E$(2)="(SYNTH.:145)"                                    <D4B>
40385 GOSUB 3001                                                  <4J6>
40390 IF XT$=CHR$(27) THEN RETURN                                 <SLO>
40395 IF XT$="(SYNTH.:145)" THEN 40325                            <7CC>
40400 NA$(0,3)=E$(1)                                              <D9J>
40405 CR%=0: RETURN                                               <DMK>
40420 EY%=21: CR%=1                                               <6TC>
40425 CHAR ,0,23,CL$: EX%=70: EL%=3: E$(2)="(SYNT
      H.:145)": E$(3)=ZA$                                         <HLJ>
40430 E$(0)="": IF NA%(0,0)>0 THEN E$(0)=MID$(STR
      $(NA%(0,0)),2,3)                                            <NTT>
40435 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN                     <UE4>
40440 IF XT$="(SYNTH.:145)" THEN CR%=2: RETURN                    <GGE>
40445 NA%(0,0)=VAL(E$(1)): CX%=0                                  <TEB>
40450 EX%=75: EL%=4: CHAR ,0,23,CL$                               <FEP>
40451 PRINT " (RVSON) I (RVOFF) - EIN-/AUSSCHALTE
      N INFORMELLE ELO-ZAHL"                                      <93T>
40452 IF CX%=1 THEN PRINT " INFORMELLE ELO-ZAHL (
      WIRD IN INGO UMGERECHNET)!";                                <CN9>
40455 E$(0)="": E$(2)="I(SYNTH.:145)": IF NA%(0,1
      )>0 THEN E$(0)=MID$(STR$(NA%(0,1)),2,4)                     <DBB>
40460 GOSUB 3001: IF XT$=CHR$(27) THEN RETURN                     <PUQ>
40465 IF XT$="(SYNTH.:145)" THEN 40425                            <5S6>
40466 IF XT$="I" THEN CX%=ABS(CX%-1): GOTO 40450                  <J5R>
40470 NA%(0,1)=VAL(E$(1)): IF CX%=1 THEN NA%(0,0)
      =INT((2840-NA%(0,1)+.5)/8): NA%(0,1)=0: NA%
      (0,0)=NA%(0,0)*SGN(SGN(NA%(0,0))+1)                         <VJM>
40475 CR%=0: CHAR ,0,23,CL$: RETURN                               <KB9>
40490 CHAR ,0,21,CR$: M1$=" / SPIELER LÖSCHEN": G
      OSUB 10011                                                  <0G6>
40495 CHAR ,0,23,CL$: PRINT " GEBEN SIE DIE SPIEL
      ERNUMMER DES ZU LÖSCHENDEN SPIELERS EIN!":
      PRINT " (RVSON) A (RVOFF) - ALLE SPIELER LÖ
      SCHEN!";                                                    <IGC>
40500 EY%=21: EX%=1: E$(0)="": E$(2)="A": E$(3)=Z
      A$                                                          <5SL>
40505 GOSUB 3001: IF XT$="A" THEN SP%=256: GOTO 4
      0525                                                        <BGU>
40510 IF XT$=CHR$(27) THEN RETURN                                 <TLO>
40515 SP%=VAL(E$(1)): IF SP%=0 OR SP%>MS% THEN GO
      SUB 2006: GOTO 40505                                        <T05>
40520 CHAR ,0,21,CR$: GOSUB 40101                                 <0A1>
40525 CHAR ,0,23,CL$: PRINT " SPIELER LÖSCHEN? ";
      JN$                                                         <2V9>
40530 GET KEY XT$: IF XT$<>"J" THEN 40490                         <0ND>
40531 IF SP%=256 THEN MS%=0: GOTO 40535                           <0JC>
40532 MS%=MS%-1                                                   <7V9>
40535 IF MS%=0 THEN WINDOW 0,5,79,20,1: PRINT "(2
      HOME)": RETURN                                              <SMR>
40540 FOR IJ=SP% TO MS%                                           <44P>
40545 NA$(IJ,0)=NA$(IJ+1,0): NA$(IJ,1)=NA$(IJ+1,1
      ): NA$(IJ,2)=NA$(IJ+1,2): NA$(IJ,3)=NA$(IJ+
      1,3): NA%(IJ,0)=NA%(IJ+1,0): NA%(IJ,1)=NA%(
      IJ+1,1)                                                     <5P8>
40550 NEXT IJ: SR%(0)=MS%: FOR IJ=1 TO MS%: SR%(I
      J)=IJ: NEXT IJ: IF SP%-7>0 THEN IJ=SP%-7: E
      LSE IJ=1                                                    <KUJ>
40555 GOSUB 40145: GOTO 40490                                     <2HE>
40570 M1$=" / SPIELERNEUEINGABE": GOSUB 10011                     <MM6>
40575 CHAR ,0,23,CL$                                              <SJN>
40580 IJ=MS%+1                                                    <FBC>
40585 DO                                                          <HA1>
40590 NA$(0,0)="": NA$(0,1)="": NA$(0,2)="": NA$(
      0,3)=""                                                     <2MV>
40595 NA%(0,0)=0: NA%(0,1)=0                                      <AD3>
40600 GOSUB 40295                                                 <273>
40605 IF CR%=1 THEN EXIT                                          <NNQ>
40610 GOSUB 40420                                                 <2B6>
40615 IF CR%=2 THEN GOSUB 40380: GOTO 40605                       <V64>
40620 IF CR%=1 THEN EXIT                                          <7NQ>
40625 NA$(IJ,0)=NA$(0,0): NA$(IJ,1)=NA$(0,1): NA$
      (IJ,2)=NA$(0,2): NA$(IJ,3)=NA$(0,3)                         <5E7>
40630 NA%(IJ,0)=NA%(0,0): NA%(IJ,1)=NA%(0,1)                      <65U>
40635 WINDOW 0,5,79,20: CHAR ,0,(MM%-5),CR$: SP%=
      IJ: GOSUB 40101: PRINT : MM%=PEEK(235): PRI
      NT "(2HOME)"                                                <KU4>
40640 IJ=IJ+1                                                     <TNB>
40645 LOOP UNTIL IJ>254                                           <TTC>
40646 IF TU%=2 THEN SM%=MS%                                       <HUD>
40650 MS%=IJ-1                                                    <9KS>
40651 IF TU%=2 AND MS%>SM% THEN GOSUB 45001                       <NEA>
40655 RETURN                                                      <AB7>
41001 M1$=" / SPIELERDATEN KORRIGIEREN": GOSUB 10
      011                                                         <USS>
41010 CHAR ,0,23,CL$                                              <BRM>
41020 PRINT " (RVSON) K (RVOFF) - KOMMANDO(2SPACE
      )*(2SPACE,RVSON) T (RVOFF,2SPACE)- TURNIERV
      ERWALTUNG(2SPACE)*(2SPACE,RVSON) F (RVOFF)
      - FUNKTIONSTASTEN BELEGEN"                                  <Q1C>
41021 PRINT " (RVSON) S (RVOFF) - ABSPEICHERN";                   <UN3>
41030 CHAR ,0,21,CR$                                              <RBO>
41040 EY%=21: EX%=1: E$(0)="": EL%=3                              <JIE>
41050 E$(3)="": IF MS%>0 THEN E$(3)=ZA$                           <S7U>
41060 E$(2)="KFS": IF MS%>MR%-1 THEN E$(2)="KFTS"                 <IDU>
41070 GOSUB 3001: IF XT$=CHR$(27) THEN 44051                      <5C9>
41080 IF XT$="K" THEN GOSUB 42001: GOTO 41001                     <D91>
41085 IF XT$="F" THEN GOSUB 53001: GOTO 40001                     <2D0>
```

Listing 1. »CH-VERWALTUNG« (Fortsetzung)

```
41090 IF XT$="T" THEN 44001                          <0GP>
41095 IF XT$="S" THEN GOSUB 51031: GOTO 41001        <DL6>
41100 IJ=VAL(E$(1)): IF IJ>MS% THEN GOSUB 2006: G
      OTO 41070                                      <70L>
41101 IF IJ=0 THEN GOSUB 41300: IF CR%=1 THEN 410
      10                                             <T3V>
41120 SP%=IJ: NA$(0,0)=NA$(SP%,0): NA$(0,1)=NA$(S
      P%,1): NA$(0,2)=NA$(SP%,2): NA$(0,3)=NA$(SP
      %,3)                                           <J9B>
41130 NA%(0,0)=NA%(SP%,0): NA%(0,1)=NA%(SP%,1)       <RHU>
41140 GOSUB 40295                                    <C73>
41150 IF CR%=1 THEN 41010                            <4EQ>
41160 GOSUB 40420                                    <FB6>
41170 IF CR%=2 THEN GOSUB 40380: GOTO 41150          <DIE>
41180 IF CR%=1 THEN 41010                            <76N>
41190 SP%=IJ: NA$(SP%,0)=NA$(0,0): NA$(SP%,1)=NA$
      (0,1): NA$(SP%,2)=NA$(0,2): NA$(SP%,3)=NA$(
      0,3)                                           <MMJ>
41200 NA%(SP%,0)=NA%(0,0): NA%(SP%,1)=NA%(0,1)       <06L>
41201 IJ=5: DO WHILE IJ<20                           <0I1>
41202 CX%=0: OPEN 3,3: CHAR ,1,(IJ): YT$="": FOR
      IK=1 TO 3: GET #3,XT$: YT$=YT$+XT$: NEXT IK
      : CLOSE 3                                      <2U9>
41203 IF VAL(YT$)=SP% THEN CX%=1: CHAR ,0,(IJ),CR
      $: GOSUB 40101: EXIT                           <B0L>
41204 IJ=IJ+1: LOOP                                  <PRQ>
41205 IF CX%=1 THEN CX%=0: GOTO 41010                <UFO>
41210 WINDOW 0,5,79,20: CHAR ,0,(MM%-5),CR$: GOSU
      B 40101: PRINT : MM%=PEEK(235): PRINT "{2HO
      ME}"                                           <2HV>
41220 GOTO 41010                                     <T0V>
41300 CHAR ,0,23,CL$: CR%=0                          <RPI>
41301 EY%=21: EX%=9: EL%=35: E$(0)="": E$(3)=ZA$+
      BS$+",": E$(2)=""                              <D9R>
41302 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: RET
      URN                                            <328>
41303 IJ=1: DO WHILE IJ<=MS%                         <5GP>
41304 IF E$(1)=LEFT$(NA$(IJ,1)+", "+NA$(IJ,2),LEN
      (E$(1))) THEN EXIT                             <A92>
41305 IJ=IJ+1: LOOP                                  <JRQ>
41306 IF IJ>MS% THEN GOSUB 2006: GOTO 41302          <4T6>
41307 RETURN                                         <AV7>
42001 CHAR ,0,23,CL$                                 <4NN>
42010 PRINT " {RVSON} L {RVOFF} - SPIELERLISTE{2S
      PACE}*{2SPACE,RVSON} F {RVOFF} - SPIELER SU
      CHEN{2SPACE}*{2SPACE,RVSON} L {RVOFF} - SPI
      ELER LÖSCHEN"                                  <HMD>
42020 PRINT " {RVSON} N {RVOFF} - NEUEINGABE{2SPA
      CE}*{2SPACE,RVSON} D {RVOFF} - DATEIÜBERNAH
      ME{2SPACE}*{2SPACE,RVSON} S {RVOFF,2SPACE}S
      PIELERLISTE SORTIEREN";                        <QHP>
42030 CHAR ,0,21,CR$                                 <R7P>
42040 GET KEY XT$                                    <FQ8>
42044 XT%=INSTR("LFNLSD",XT$): IF XT%=0 THEN RETU
      RN                                             <V5K>
42045 IF XT%=2 AND MS%=254 THEN GOSUB 2006: GOTO
      42040                                          <3PJ>
42046 IF TU%=2 AND XT%>3 THEN GOSUB 2006: GOTO 42
      040                                            <HBH>
42047 IF MS%=0 AND XT%<>3 AND XT%<>6 THEN GOSUB 2
      006: GOTO 42040                                <9LT>
42048 IF MS%>0 AND XT%=6 THEN GOSUB 2006: GOTO 42
      040                                            <BBO>
42050 ON XT% GOTO 40130,40220,40570,40490,43001,4
      2100                                           <CFU>
42100 CHAR ,0,23,CL$: PRINT " DATEINAME:": EX%=13
      : EY%=23: EL%=15: E$(0)="": E$(2)="": E$(3)
      =" ABCDEFGHIJKLMNOPQRSTUVWXYZ.+-/?"+ZA$        <6M9>
42105 GOSUB 3001: IF XT$=CHR$(27) THEN 42001         <7S3>
42107 YD$="S"+LEFT$(E$(1)+"*",15)                    <1L9>
42110 DOPEN #2,(YD$)                                 <IFR>
42115 IF DS=0 THEN 42125                             <RVS>
42120 GOSUB 4001: IF CR%=0 THEN 42110: ELSE GOTO
      42100                                          <7TN>
42125 INPUT#2,MS%: IF MS%=0 THEN 42135               <FB6>
42130 FOR II=1 TO MS%: FOR IJ=0 TO 3: INPUT#2,YD$
      : NA$(II,IJ)=LEFT$(YD$,LEN(YD$)-1): NEXT IJ
      : INPUT#2,NA%(II,0): INPUT#2,NA%(II,1): NEX
      T II                                           <6NA>
42135 IF DS>0 THEN 42120                             <QJV>
42140 DCLOSE #2: GOTO 40015                          <QNR>
43001 CHAR ,0,23,CL$: PRINT " SPIELERLISTE WIRD S
      ORTIERT!"                                      <61B>
43002 BANK 0: POKE DEC("1400"),MS%                   <DCC>
43003 FOR IJ=1 TO MS%                                <MF4>
43010 FOR IK=0 TO 1                                  <BI3>
43020 XX%=NA%(IJ,IK): IF IK=0 AND XX%=0 THEN XX%=
      999                                            <0FB>
43021 IF IK=1 AND XX%=0 AND NA%(IJ,0)<130 THEN XX
      %=SGN(NA%(IJ,0))*(2839-B*NA%(IJ,0))            <40K>
43030 XY%=INT(XX%/256): XZ%=XX%-XY%*256              <TKB>
43040 POKE DEC("CF00")-IK*512+IJ,XZ%: POKE DEC("C
      E00")-IK*512+IJ,XY%                            <DIF>
43050 NEXT IK                                        <OTC>
43060 XB$=NA$(IJ,1)+CHR$(0)+NA$(IJ,2)+CHR$(13): F
      OR IL=1 TO LEN(XB$)                            <RJH>
43070 POKE DEC("D000")+(IJ-1)*37+IL-1,ASC(MID$(XB
      $,IL,1))                                       <A6V>
43080 NEXT IL                                        <BRF>
43090 NEXT IJ                                        <O1C>
43100 SYS 5376                                       <Q3M>
43105 FOR IJ=1 TO MS%: SR%(IJ)=PEEK(DEC("1400")+I
      J): NEXT IJ                                    <K71>
43110 GOSUB 51131: IF CR%=0 THEN 40134: ELSE RETU
      RN                                             <3U2>
44001 IF TU%=2 THEN CX%=1: GOTO 44051                <IO4>
44010 M1$=" >> TURNIERVERWALTUNG": GOSUB 10011       <3QS>
44020 CHAR ,0,23,CL$                                 <GFK>
44030 PRINT " SIND SIE SICHER? "JN$" - DIE SPIELE
      RLISTE KANN NUR BEDINGT VERÄNDERT": PRINT "
       WERDEN!";                                     <QHB>
44040 GET KEY XT$: IF XT$<>"J" THEN 41001            <97B>
44041 CHAR ,0,23,CL$: PRINT " SOLL SPIELERLISTE S
      ORTIERT WERDEN ? "JN$                          <CBN>
44042 GET KEY XT$: IF XT$=CHR$(27) THEN 41001        <15M>
44043 IF XT$="J" THEN 44046                          <MVS>
44044 GOSUB 51031: IF CR%=1 THEN 41001               <PLB>
44045 GOTO 44050                                     <JBF>
44046 GOSUB 43001: IF CR%=1 THEN 41001               <Q5D>
44050 BANK 1: SYS LE%: SYS LT%: POKE DEC("1000"),
      1: POKE DEC("1001"),1: GOTO 20001              <ATB>
44051 CHAR ,0,23,CL$: PRINT DL$;JN$: GET KEY XT$:
       IF XT$="N" THEN 44070                         <ARJ>
44052 IF XT$=CHR$(27) THEN CX%=0: GOTO 41001         <69V>
44060 GOSUB 51031: IF CR%=1 THEN 41001               <PLB>
44070 ON CX% GOTO 20001: GOTO 63901                  <2SF>
45001 BANK 1: RD%=1                                  <D16>
45005 DO                                             <78P>
45010 FOR IJ=SM%+1 TO MS%                            <TIK>
45020 POKE DEC("1000")+IJ*16+RD%,0                   <B0B>
45030 POKE DEC("2000")+IJ*16+RD%,16                  <L0A>
45040 NEXT IJ                                        <9BC>
45050 IF PEEK(DEC("1001")+RD%)=0 OR RD%=15 THEN E
      XIT                                            <E6C>
45060 RD%=RD%+1: LOOP                                <SB0>
45070 RETURN                                         <3Q7>
50001 CHAR ,0,23,CL$                                 <4NN>
50005 PRINT " GEBEN SIE DEN DATEINAMEN EIN:"         <28T>
50010 EX%=31: EL%=13: EY%=23: E$(0)=FI$: E$(2)=""
      : E$(3)=ZA$+"-+/.ABCDEFGHIJKLMNOPQRTSUVWXYZ
      ": IF CA%=1 THEN E$(3)=E$(3)+"?"               <B9K>
50015 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: GOT
      O 50017: ELSE CR%=0                            <7TQ>
50016 FI$=E$(1)                                      <BQV>
50017 CHAR ,0,23,CL$: RETURN                         <BSL>
50031 YD$="?"+FI$: DOPEN #2,(YD$): IF DS=62 THEN
      CR%=0: DCLOSE #2: RETURN                       <0QB>
50032 DCLOSE #2                                      <695>
50033 GOSUB 4001: IF CR%=0 THEN 50031: ELSE RETUR
      N                                              <0TJ>
50041 CHAR ,0,23,CL$                                 <4JN>
50046 PRINT " GEBEN SIE DEN TURNIERNAMEN UND GGF.
       DIE ZU SPIELENDE RUNDENZAHL EIN:"             <BRD>
50051 PRINT " {RVSON}NAME:{RVOFF}";TAB(69)"{RVSON
      }RD.:{RVOFF}";                                 <P0C>
50056 EY%=24                                         <4EU>
50060 EX%=7: EL%=60: E$(0)=TU$: E$(2)="": E$(3)=B
      S$+ZA$+CHR$(160)                               <EID>
50065 GOSUB 3001: IF XT$=CHR$(27) THEN CR%=1: GOT
      O 50110                                        <F39>
50070 TU$=E$(1): IF TU$="" THEN 50065                <7GQ>
50071 IF TU%=2 THEN 50100                            <M2O>
50075 EX%=74: EL%=2: E$(2)="{SYNTH.:145}": E$(3)=
      ZA$                                            <MJC>
50076 E$(0)="": IF MR%>0 THEN E$(0)=MID$(STR$(MR%
      ),2,2)                                         <59L>
50080 GOSUB 3001: IF XT$="{SYNTH.:145}" THEN 5006
      0                                              <AUH>
50085 IF XT$=CHR$(27) THEN CR%=1: GOTO 50110         <6BT>
50090 MR%=VAL(E$(1)): IF MR%<5 OR MR%>15 THEN 500
      80                                             <R6P>
50100 CR%=0                                          <M3V>
50110 CHAR ,0,23,CL$: RETURN                         <DSD>
51001 GOSUB 4131: IF CR%=1 THEN RETURN               <A7B>
51002 YD$="K"+FI$: SCRATCH(YD$)                      <3H6>
51003 DOPEN #2,(YD$),W: IF DS=0 THEN 51005           <LKE>
51004 GOSUB 4001: IF CR%=0 THEN 51001: ELSE RETUR
      N                                              <AP3>
51005 PRINT#2,FI$: PRINT#2,TU$: PRINT#2,TU%: PRIN
      T#2,MR%: IF DS>0 THEN 51004: ELSE DCLOSE #2
      : CR%=0: RETURN                                <E7H>
51011 GOSUB 4131: IF CR%=1 THEN RETURN               <87B>
51012 YD$="T"+FI$: SCRATCH(YD$)                      <C8O>
51013 BSAVE(YD$) ON B1,P(DEC("1000")) TO P(DEC("3
      000"))                                         <UF7>
51014 IF DS=0 THEN CR%=0: RETURN                     <NPF>
51015 GOSUB 4001: IF CR%=0 THEN GOTO 51011: ELSE
      RETURN                                         <SI5>
51031 FOR II=1 TO MS%: SR%(II)=II: NEXT II           <7NV>
51032 GOSUB 4131: IF CR%=1 THEN RETURN               <F7B>
51033 YD$="S"+FI$: SCRATCH(YD$): DOPEN #2,(YD$),W
      : IF DS=0 THEN 51035                           <D47>
51034 GOSUB 4001: IF CR%=0 THEN 51032: ELSE RETUR
      N                                              <NQ3>
51035 PRINT#2,MS%: IF MS%=0 THEN 51037               <0N1>
51036 FOR II=1 TO MS%: FOR IJ=0 TO 3: PRINT#2,NA$
      (SR%(II),IJ)"←": NEXT IJ: PRINT#2,NA%(SR%(I
      I),0): PRINT#2,NA%(SR%(II),1): NEXT II         <21D>
51037 IF DS>0 THEN 51034                             <R7P>
51038 DCLOSE #2: CR%=0: RETURN                       <2J3>
51131 GOSUB 51032: IF CR%=1 THEN RETURN : ELSE GO
      TO 52001                                       <LNR>
52001 GOSUB 4131: IF CR%=1 THEN RETURN               <I7B>
52002 YD$="S"+FI$: DOPEN #2,(YD$)                    <K0B>
52003 IF DS=0 THEN 52005                             <OVG>
52004 GOSUB 4001: IF CR%=0 THEN 52001: ELSE RETUR
      N                                              <IL3>
52005 INPUT#2,MS%: IF MS%=0 THEN 52007               <DFE>
52006 FOR II=1 TO MS%: FOR IJ=0 TO 3: INPUT#2,YD$
      : NA$(II,IJ)=LEFT$(YD$,LEN(YD$)-1): NEXT IJ
      : INPUT#2,NA%(II,0): INPUT#2,NA%(II,1): NEX
      T II                                           <63A>
52007 IF DS>0 THEN 52004                             <P3H>
52008 DCLOSE #2: CR%=0: RETURN                       <QS3>
52011 GOSUB 4131: IF CR%=1 THEN RETURN               <G7B>
52012 YD$="T"+FI$: OPEN 2,8,2,YD$+",P,R": IF DS=0
       THEN CLOSE 2: GOTO 52014                      <5SV>
52013 GOSUB 4001: IF CR%=0 THEN 52011: ELSE RETUR
      N                                              <4NJ>
52014 BLOAD(YD$) ON B1,P(DEC("1000")): RETURN        <AFR>
53001 M1$=" / FUNKTIONSTASTEN BELEGEN": GOSUB 100
      11                                             <G29>
53005 CHAR ,0,3,CL$: WINDOW 1,4,79,22,1: KEY : PR
      INT "{2HOME}"                                  <94E>
53010 CHAR ,0,23: PRINT " GEWÜNSCHTE BELEGUNG: ({
      RVSON} COM-E {RVOFF} = {RVSON} ESC {RVOFF,2
      SPACE}*{2SPACE,RVSON} COM-R {RVOFF} = {RVSO
      N} RTN {RVOFF})"                               <QSG>
53015 IJ=1: EX%=7: EL%=30: EY%=24                    <AV6>
53020 DO                                             <NQR>
53025 E$(3)=ZA$+BS$+","+"£T": E$(2)="{SYNTH.:145,
      DOWN}": E$(0)=""                               <PD8>
53030 CHAR ,0,24,CR$: PRINT " {RVSON} F" RIGHT$(S
      TR$(IJ),1)" {RVOFF}:";                         <1BI>
53035 GOSUB 3001: IF XT$=CHR$(27) THEN EXIT          <L4R>
53040 IF XT$="{SYNTH.:145}" THEN IJ=IJ-1: GOTO 53
      085                                            <KHB>
53045 IF XT$="{DOWN}" THEN 53080                     <AHR>
53050 DO                                             <GQ7>
```

Listing 1. »CH-VERWALTUNG« (Fortsetzung)

64'er SONDERHEFT

PROGRAMM-SERVICE

Direkt bestellen statt abtippen!

Die aktuelle Diskette zum Heft:

64'er Sonderheft 36:

Programme für alle C128-Besitzer

Haushaltsbuch: die Finanzen im Griff
Bringen Sie Ordnung in Ihre Finanzen mit dem Haushaltsbuch für den C128. Durch die Kontrolle über Ihre Ausgaben und gezieltes Wirtschaften, bleibt Ihnen sicher noch etwas über für »die kleinen Annehmlichkeiten des Lebens«.

Professionelle Schachturnierverwaltung
Ein Profi-Programm, das schon bei mancher Schachmeisterschaft, so zum Beispiel den 15. Dortmunder Schachtagen, eingesetzt wurde. Bis zu 254 Spieler können nach dem »Schweizer System« verwaltet werden.

Power 128
Ein universelles Disketten-Tool, das den Eigenheiten der Floppy 1571 angepaßt ist. Sie können das Directory beliebig manipulieren, sortieren und sich als Zugabe eine Sicherheitskopie des Directorys anlegen, die dem »Read Error 18 01« seine fatalen Folgen nimmt.

Spiel-Spaß total: Super-Vectors
Vectors, dem Motorrad-Rennen aus dem Film »Tron« nachempfunden, erleben Sie hier in vollkommen neuen Grafik-Dimensionen. Die Bildschirmauflösung beträgt nun 736x354 Punkte. Dies ist mehr als professionelle PCs mit einer Hercules-Grafikkarte darstellen können. Speziell für den C128D im Blechgehäuse mit 64 KByte Video-RAM und alle nachträglich aufgerüsteten C128 wurde dieses Spiel umgeschrieben.
Weiterhin befinden sich auf der Diskette alle Programme, die im Inhaltsverzeichnis des 64'er-Sonderhefts 36 mit einem Diskettensymbol gekennzeichnet sind.

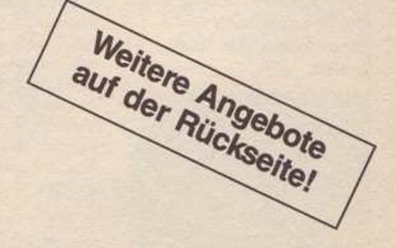
5¼"-Diskette für C64/C128
Bestell-Nr. 15836

DM 29,90* *(sFr 24,90*/öS 299,–*)*
*** Unverbindliche Preisempfehlung**

Wenn Sie Fragen zu diesen Programmen oder zu anderen Angeboten aus unserem Programm-Service haben, rufen Sie uns an:
Telefon (089) 4613-640

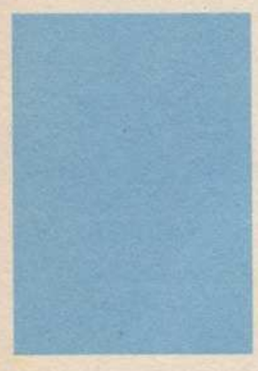






Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8012 Haar bei München, Telefon (089) 4613-0





Sie suchen packende Spiele, hilfreiche Utilities und professionelle Anwendungen für Ihren Computer? Sie wünschen sich gute Software zu vernünftigen Preisen? Hier finden Sie beides! Unser stetig wachsendes Sortiment enthält interessante Listing-Software für alle gängigen Computertypen. Jeden Monat erweitert sich unser aktuelles Angebot um eine weitere interessante Programmsammlung für jeweils einen Computertyp.
Wenn Sie Fragen zu den Programmen in unserem Angebot haben, rufen Sie uns an:
Telefon (089) 4613-640

Bestellungen bitte *nur gegen Vorauskasse* an: Markt&Technik Verlag AG, Unternehmensbereich Buchverlag, Hans-Pinsel-Straße 2, D-8013 Haar, Telefon (089) 4613-0.
Schweiz:
Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656.
Österreich:
Microcomput-ique, E. Schiller, Fasangasse 24, A-1030 Wien, Telefon (0222) 785661;
Bücherzentrum Meidling, Schönbrunner Straße 261, A-1120 Wien, Telefon (0222) 833196.
Bestellungen aus anderen Ländern bitte nur schriftlich an: Markt&Technik Verlag AG, Abt. Buchvertrieb, Hans-Pinsel-Straße 2, D-8013 Haar, und gegen Bezahlung der Rechnung im voraus.

Bitte verwenden Sie für Ihre Bestellung und Überweisung die abgedruckte Postgiro-Zahlkarte, oder senden Sie uns einen Verrechnungs-Scheck mit Ihrer Bestellung. Sie erleichtern uns die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

64'er Sonderheft 35: Assembler

Hypra-Ass: Ein Makro-Assembler der Spitzenklasse. Er erlaubt es, Maschinensprache-Programme ähnlich komfortabel wie in Basic zu schreiben. Durch Makros – dies sind immer wieder benötigte Unterprogramme, die über ihren Namen aufgerufen werden – und bedingte Assemblierung ist große Übersichtlichkeit auch bei langen Programmen gewährleistet. Ein weiterer Vorteil ist der Editor des Hypra-Ass, dessen formatierende LIST-Routine größtmögliche Übersicht am Bildschirm gewährleistet. **Reass:** Quasi die Umkehrung des Hypra-Ass und mindestens genauso wichtig ist dieser Reassembler. Sie haben beispielsweise ein Maschinenprogramm bekommen, das Ihnen gut gefällt, bis eben auf einige Kleinigkeiten, die Sie ändern wollen. Doch wie? Im Dickicht des reinen Maschinencodes findet sich niemand zurecht und ein Quellcode ist nicht vorhanden. Hier hilft der Reass: Er macht aus dem Maschinenprogramm wieder gut lesbaren, strukturierten Quellcode, der mit Labels und Zeilennummern versehen direkt mit dem Hypra-Ass bearbeitet werden kann. **SMON:** Dieser leistungsfähige Speichermonitor erlaubt Ihnen, bis in die tiefsten Tiefen Ihres Computers vorzudringen. Mit dem SMON lassen sich die Prozessorregister anzeigen und beeinflussen, Speicherbereiche anzeigen, vergleichen, verschieben und... und... und. Integriertes Disassemblieren erlaubt das Entschlüsseln von Maschinencode. Mit dem Miniassembler können Sie »mal eben« ein kleines Programm eingeben. Der Trumpf des SMON ist der ebenfalls integrierte Diskettenmonitor, der Ihnen zusätzlich volle Kontrolle über die Floppystation gibt. **Kurse:** Zwei komplette und leichtverständliche Kurse bringen selbst Assembler-Einsteiger auf das Profi-Niveau. Alle Programm-Routinen, die in den Kursen ausführlich erläutert werden, finden Sie ebenfalls auf dieser Programmservice-Diskette. Weiterhin befinden sich auf der Diskette alle Programme, die im Inhaltsverzeichnis des 64'er-Sonderhefts 35 mit einem Diskettensymbol gekennzeichnet sind. Diskette für C64/C128

Bestell-Nr. 15835 **DM 19,90*** sFr 17,–*/öS 199,–*

64'er Sonderheft 34: Aufbruch in die dritte Dimension!

Fantastische Perspektiven: Ein Grafikprogramm mit ganz neuen Leistungsmerkmalen. Das Konstruieren von perspektivischen Grafiken, wie etwa ganzen Straßenzügen, wird zum Vergnügen. **Ein Freezer für harte Fälle:** Auf Knopfdruck wird der gesamte Speicher des C64 auf Diskette gespeichert und bei Bedarf wieder geladen. Sie arbeiten an der gleichen Stelle weiter, als wäre nichts geschehen. Sicherheitskopien von kopiergeschützten Originaldisketten anfertigen, Spielstände bei hartnäckigen Games »einfrieren« und bei Bedarf wieder laden – all das sind Stärken des Freezers. **3D-Movie-Maker:** Das Konstruieren von dreidimensionalen Körpern und deren fließende Bewegung am Bidschirm zeichnet dieses Programm aus. Nach der Eingabe der Koordinaten für den Körper und seine Bewegung erzeugen Sie faszinierende Trickfilme in 3D. **Perfekte Simulation:** Selten zuvor wurde so anschaulich gezeigt, wie eine Braunsche Röhre – Urahn aller Bildschirme und Monitore – funktioniert. Per Tastendruck steuern Sie alle Parameter. Veränderungen des Elektronenstrahls werden in Echtzeit am Bildschirm angezeigt. **Kurvendiskussion perfekt:** Ein Segen für alle, die sich mit Mathematik und vor allem der Kurvendiskussion beschäftigen. Nicht nur, daß jede Funktion am Bildschirm anschaulich dargestellt wird. Auch die Ableitungen, Nullstellen etc. werden sofort berechnet und ausgegeben. **Digital einfach:** Ideal für den Einstieg in die Digital-Elektronik ist dieses Programm. Alle Grundfunktionen, wie AND-, OR-, EXOR-Gatter etc. werden am Bildschirm dargestellt und in ihren Funktionen simuliert. Weiterhin befinden sich auf der Diskette alle Programme, die im Inhaltsverzeichnis des 64'er-Sonderhefts 34 mit einem Diskettensymbol gekennzeichnet sind. Diskette für C64/C128

Bestell-Nr. 15834 **DM 29,90*** sFr 24,90*/öS 299,–*

64'er Sonderheft 33: Tips, Tricks&Tools zum C64/C128

Titelgenerator: Wenn Sie schon immer professionelle Programmvorspänne für Ihre eigenen Programme verwenden wollten: Jetzt können Sie es. Ohne Programmierkenntnisse erstellen Sie faszinierende Vorspänne mit Musik, Laufschrift, eigenen Zeichensätzen, Grafiken und... und... und. **Hypra-Sprite:** Ein Sprite-Editor der Spitzenklasse, der Funktionen bietet, die auf diesem Gebiet noch nicht da waren. Mit einer komfortablen Benutzeroberfläche lassen sich Sprites erzeugen, dehnen, stauchen, spiegeln etc. Das Erzeugen von Animationssequenzen und kleinen Filmen gehört zu den Leckerbissen des Programms. **Basic-Kontroll-System:** Machen Sie Schluß mit Syntaxfehlern und unsauberem Programmierstil in Ihren Basic-Programmen! Das Basic-Kontroll-System spürt die häufigsten Programmfehler auf und hebt so das Qualitätsniveau Ihrer Software. **Alpha-Drummer:** Der C64 wird zum Super-Schlagzeuger. 24 digitalisierte Drum-Sounds stehen Ihnen sofort zur Verfügung. Weitere können mit der im Programm integrierten Digitizer-Software von Ihnen erzeugt und bearbeitet werden. Stücke mit einer Länge von über 15 Minuten sind kein Problem. **Apfelsee:** Eine ganz neue Variante der Fraktalprogramme: Entdecken Sie die bizarre Schönheit dreidimensionaler Fraktal-Landschaften. Weiterhin befinden sich auf der Diskette alle Programme, die im Inhaltsverzeichnis des 64'er-Sonderhefts 33 mit einem Diskettensymbol gekennzeichnet sind. Diskette für C64/C128

Bestell-Nr. 15833 **DM 29,90*** sFr 24,90*/öS 299,–*

64'er Sonderheft 29: Programme, die jeder C128-Besitzer braucht

Mastertext 128: Die Super-Textverarbeitung für den 80-Zeichen-Modus mit eingebauter Rechtschreibprüfung. Komfort und Funktionsvielfalt werden bei diesem Programm großgeschrieben. Alle Standardbefehle der modernen Textverarbeitung, ein integrierter Taschenrechner und sogar der Datenaustausch per DFÜ sind enthalten. Als besonderen Leckerbissen bietet Mastertext 128 eine Rechtschreibprüfung, deren Wörterbuch beliebig erweiterbar ist. Tippfehler gehören damit der Vergangenheit an! **Der Hexer:** Endlich ein leistungsstarkes Kopierprogramm für den C128. Kopieren Sie nach Herzenslust, der Hexer wird auch Ihre Disketten bezaubern. Neben ganzen Disketten sind mit diesem Programm auch einzelne Files zu kopieren. Der Bedienkomfort des Hexers ist kaum zu überbieten. Probleme mit den verschiedenen Versionen des C128 kennt der Hexer nicht, es stehen verschiedene Versionen »für alle Fälle« bereit. Besitzer des »Dolphin-DOS« können sich über eine Version freuen, die mit diesem Floppy-Beschleuniger zusammenarbeitet. **Unidat Pro:** Der Wunsch jedes ernsthaften Computer-Anwenders ist eine leistungsfähige und komfortable Dateiverwaltung. Mit Unidat Pro wird dieser Wunsch Realität. Erstellen und verändern Sie eigene Datei-Masken. Hohe Zugriffsgeschwindigkeit auf Ihre Daten, die Unterstützung von Paßwörtern zum Datenschutz und eine Export-Funktion zeichnen diese Dateiverwaltung aus. Die Suche nach einem Datensatz erfolgt blitzschnell. **Mancomania:** Spielen Sie gerne Wirtschaftsspiele? Wenn Ihnen diese Spielgattung gefällt, ist Mancomania das Richtige für Sie. Das Spielziel ist allerdings ein wenig anders als bei den üblichen Vertretern dieses Genres: Verschleudern Sie Ihr Vermögen, so schnell Sie können. Vertreiben Sie sich die Zeit im Spiel-Casino, kaufen Sie Aktien an der Börse, und wetten Sie beim Autorennen. Denken Sie daran, das Geld muß weg. Aber das ist leichter gesagt als getan, als Millionär hat man's halt schwer! Diskette für C128

Bestell-Nr. 15829 **DM 29,90*** sFr 24,90*/öS 299,–*

64'er Sonderheft 28: Programme und Utilities zu GEOS

Geoterm: Erschließen Sie sich die Welt der Datenfernübertragung mit GEOS. Geoterm ist ein Terminalprogramm der Spitzenklasse. Alle Funktionen sind wie von GEOS gewohnt mit Maus und Pull-down-Menüs steuerbar. So leicht war DFÜ noch nie. Sie wollen Ihre Grafiken, die Sie im Hi-Eddi, Koalapainter-, Doodle-Format etc. vorliegen haben, in Geo-Write, GeoPaint GeoFile verwenden? Kein Problem, der **Bitmap-Converter** macht's möglich. Das Programm arbeitet vollständig unter GEOS und speichert Ihre Grafiken im Format von GEOS-Foto-Scraps. Diese können mit nahezu jedem GEOS-Programm weiterverarbeitet werden. Ärger mit dem Drucker? Erstklassige Qualität erhalten Ihre Ausdrucke unter GEOS mit den verschiedensten Druckertreibern für den Star NL-10, Epson-Drucker und Kompatible und den Citizen 120D. Mit **Superprint V2.0** läßt sich zudem nahezu jeder störrische Drucker an GEOS anpassen. **GEOS-Icon-Editor** und **GEOS-Pattern-Editor:** Zwei Programme, die in keiner GEOS-Programmsammlung fehlen dürfen. Sie erlauben es, eigene Programme mit Icons (Piktogrammen) zu versehen. Mit dem Pattern-Editor kann jeder seine eigenen Füllmuster für GeoPaint nach Wunsch definieren. Eine Seite der Diskette wird im GEOS-Format ausgeliefert. Alle GEOS-Programme sind ohne Zusatzaufwand unter GEOS sofort lauffähig. **Datec:** Ein Datenverwaltungsprogramm der Superlative (kein GEOS-Programm!). Freie Dateneingabe- und Druckmasken (beispielsweise für Etiketten) sind definierbar. Umfangreiche Such- und Indexfunktionen sowie frei definierbare Zeichensätze (natürlich mit deutschen Umlauten) sind nur einige der Glanzpunkte dieses Programms. Diskette für C64/C128

Bestell-Nr. 15828 **DM 29,90*** sFr 24,90*/öS 299,–*

64'er Sonderheft 27: Ein unglaubliches Multicolor-Mal- und -Zeichenprogramm

Amica-Paint: Dieses Programm bietet Funktionen, die man vorher nur dem Amiga zugetraut hatte: Amica-Paint dreht, kippt und spiegelt beliebige Bildausschnitte und berechnet selbständig Farbverläufe. Definition von Makros, eine eingebaute Diashow-Funktion und natürlich komfortable Maussteuerung sind nur einige wenige Features dieses absolut sensationellen Programms. **Schreibmaschine:** Entlocken Sie Ihrem Drucker Lettern in einer Qualität und Schönheit, die Sie ihm nicht zugetraut hätten. Viele Schriftarten befinden sich aus Platzgründen nur auf Diskette. **Pic-Change:** Darauf haben die Grafik-Fans schon lange gewartet: Ob HiRes oder Mulicolor – Pic-Change macht Schluß mit dem Wirrwarr verschiedener Grafikformate. Jedes übliche Grafikformat kann in jedes andere übersetzt werden. **Grafik 2001:** Eine leistungsfähige Erweiterung zur Basic-Erweiterung »Grafik 2000« mit vielen neuen Befehlen. Grafik 2000 (aus dem Sonderheft 4) ist auf der Programmservice-Diskette ebenfalls enthalten. Weiterhin finden Sie alle Programme auf Diskette, die im Inhaltsverzeichnis mit einem Diskettensymbol gekennzeichnet sind. Zwei Disketten für C64/C128

Bestell-Nr. 15827 **DM 34,90*** sFr 29,50*/öS 349,–*

* Unverbindliche Preisempfehlung

Übrigens: Mit den Gutscheinen aus dem »Super-Software-Scheckheft« für DM 149,– können Sie sechs Software-Disketten Ihrer Wahl aus dem Programm-Service-Angebot der Zeitschriften

PC Magazin	Happy-Computer-Sonderheft	Computer persönlich
PC Magazin Plus	Amiga-Magazin	64'er-Magazin
Happy-Computer	Amiga-Sonderheft	64'er-Sonderheft

bestellen – egal, ob diese DM 29,90 oder DM 34,90 kosten. Das Scheckheft können Sie per Verrechnungsscheck oder mit der eingehefteten Zahlkarte direkt beim Verlag bestellen.
Kennwort: Software-Scheckheft, Bestell-Nr. 39100.

```
53055 XT%=INSTR(E$(1),"£"): IF XT%>0 THEN E$(1)=L
      EFT$(E$(1),XT%-1)+CHR$(13)+MID$(E$(1),XT%+1
      ,80)                                           <IJN>
53056 IF XT%=0 THEN EXIT                             <M7S>
53057 LOOP                                           <SJ1>
53058 DO                                             <KQN>
53060 XT%=INSTR(E$(1),"T"): IF XT%>0 THEN E$(1)=L
      EFT$(E$(1),XT%-1)+CHR$(27)+MID$(E$(1),XT%+1
      ,80)                                           <EG7>
53065 IF XT%=0 THEN EXIT                             <6FT>
53070 LOOP                                           <QJP>
53075 KEY IJ,E$(1): WINDOW 1,4,79,22,1: KEY : PRI
      NT "{2HOME}"                                   <E99>
53080 IJ=IJ+1: IF IJ=9 THEN IJ=1                     <9E0>
53085 IF IJ=0 THEN IJ=8                              <SBV>
53090 LOOP                                           <2KP>
53095 RETURN                                         <A37>
63500 OPEN 2,8,2,"P2.1,P,R": GET #2,XD$              <HTV>
63501 IF DS=0 THEN CLOSE 2: RUN "P2.1"               <GSS>
63502 GOTO 63602                                     <IGB>
63600 OPEN 2,8,2,"P1.1,P,R": GET #2,XD$              <H9T>
63601 IF DS=0 THEN CLOSE 2: RUN "P1.1"               <LSG>
63602 CLOSE 2: GOSUB 2006: CHAR ,0,23,CL$            <BV1>
63603 PRINT " PROGRAMMDISKETTE EINLEGEN!"            <N79>
63604 GET KEY XT$: IF XT$=CHR$(13) THEN 63600: EL
      SE GOTO 63901                                  <T0T>
63701 XX$=FI$: XY$=DN$: XZ$=ID$: XW$=TU$: XX%=TU%
      : XZ%=MR%                                      <BNF>
63702 CA%=1: GOSUB 50001: CA%=0                      <PQN>
63705 IF CR%=1 THEN FI$=XX$: DN$=XY$: ID$=XZ$: TU
      $=XW$: TU%=XX%: MR%=XZ%: GOTO 63935            <DA3>
63715 YD$="X"+FI$+"*"                                <73H>
63720 DOPEN #2,(YD$): IF DS=0 THEN 63730             <001>
63725 GOSUB 4001: IF CR%=1 THEN 63702: ELSE GOTO
      63720                                          <MBB>
63730 INPUT#2,FI$: INPUT#2,TU$: INPUT#2,TU%: INPU
      T#2,MR%: IF DS>0 THEN 63725                    <0SH>
63735 DCLOSE #2                                      <055>
63740 GOSUB 4121                                     <ERQ>
63750 IF CR%=1 THEN 63702                            <4UO>
63755 GOSUB 52001                                    <OFC>
63760 IF CR%=1 THEN 63702                            <46R>
63765 IF TU%=2 THEN GOSUB 52011: IF CR%=1 THEN 63
      702                                            <339>
63770 GOSUB 10005: GOTO 63993                        <F51>
63801 XX$=FI$: XY$=DN$: XZ$=ID$: XW$=TU$: XX%=TU%
      : XZ%=MR%                                      <BF0>
63802 TU%=0: GOSUB 50041                             <2UT>
63805 IF CR%=1 THEN FI$=XX$: DN$=XY$: ID$=XZ$: TU
      $=XW$: TU%=XX%: MR%=XZ%: GOTO 63935            <L53>
63810 CA%=0: GOSUB 50001                             <I6U>
63815 IF CR%=1 THEN 63802                            <5AU>
63816 IF FI$="" THEN GOSUB 2006: GOTO 63810          <KRF>
63820 GOSUB 50031: IF CR%=1 THEN 63810               <9TA>
63822 GOSUB 4121: IF CR%=1 THEN 63810                <F4H>
63825 TU%=1: GOSUB 51001: IF CR%=1 THEN 63810        <CMT>
63830 MS%=0: GOSUB 10005: GOTO 63993                 <FQ5>
63901 CX%=0: M0$="HAUPTMENUE": M1$="": DR%=0: GOS
      UB 10005                                       <048>
63905 CHAR ,0,3,CL$                                  <4D3>
63915 PRINT "{DOWN} {RVSON,2SPACE}1{2SPACE,RVOFF}
       - TURNIER VON DISKETTE                        <T2F>
63920 PRINT "{DOWN} {RVSON,2SPACE}2{2SPACE,RVOFF}
       - NEUES TURNIER                               <U1L>
63925 PRINT "{DOWN} {RVSON,2SPACE}3{2SPACE,RVOFF}
       - DIENST (FUNKTIONSTASTEN BELEGEN, DISKETT
      ENVERWALTUNG, ETC.)                            <1F3>
63926 PRINT "{DOWN} {RVSON,2SPACE}4{2SPACE,RVOFF}
       - KOPFDATEN ANDERN                            <QQT>
63927 PRINT                                          <QEE>
63929 PRINT "{DOWN} {RVSON} RTN {RVOFF} - WEITER
      OHNE ANDERUNG                                  <JUK>
63930 PRINT "{DOWN} {RVSON} ESC {RVOFF} - PROGRAM
      M BEENDEN                                      <P04>
63935 YT$="1234"+CHR$(13)+CHR$(27)                   <Q59>
63940 GET KEY XT$                                    <CQ8>
63941 XT%=INSTR(YT$,XT$)                             <KF9>
63945 ON XT% GOTO 63701,63801,30001,63975,63975,6
      3950: GOTO 63940                               <SU3>
63950 CHAR ,0,23,CL$                                 <ONN>
63960 PRINT " WOLLEN SIE DAS PROGRAMM WIRKLICH BE
      ENDEN? ";JN$: GOSUB 2001                       <U4P>
63965 GET KEY XT$: IF XT$="J" THEN CLR : PRINT "{
      2HOME,CLR}": GOTO 63999                        <ND4>
63970 CHAR ,0,23,CL$: GOTO 63940                     <7E7>
63975 IF TU%=0 THEN 63980                            <KIG>
63976 IF XT%=7 THEN 63991: ELSE GOTO 63994           <7I9>
63980 GOSUB 2006                                     <071>
63985 CHAR ,0,23,CL$                                 <4RM>
63990 PRINT " ERST TURNIER EINLADEN!": GET KEY XT
      $: GOTO 63941                                  <IOS>
63991 GOSUB 52001: IF CR%=1 THEN 63935               <VPU>
63992 IF TU%=2 THEN GOSUB 52011: IF CR%=1 THEN 63
      935                                            <D6L>
63993 ON TU% GOTO 40001,20001                        <NIB>
63994 XX$=TU$: XX%=TU%: XY%=MR%: GOSUB 50041: IF
      CR%=1 THEN TU$=XX$: TU%=XX%: MR%=XY%: GOTO
      63940                                          <NSS>
63995 GOSUB 51001: IF CR%=1 THEN 63994               <VDC>
63996 GOTO 63901                                     <SSK>
63999 BANK 15: SYS 65341                             <UB4>
```

Listing 1. »CH-VERWALTUNG« (Schluß)

```
8 GRAPHIC CLR : GRAPHIC 5,1: FAST : POKE 0,PEEK(0
  ) OR 64: POKE 1,0: COLOR 6,1: PRINT "{GREY3,CTR
  L+N,CTRL+K}";                                      <VR5>
9 SLEEP 2: TRAP 1000                                 <5AV>
10 PRINT "{2HOME,CLR}CCCCCC MM{2SPACE}MM EEEEEE S
   SSSSS SSSSSS                                      <DVS>
20 PRINT "CC{5SPACE}MM{2SPACE}MM EE{5SPACE}SS{5SP
   ACE}SS                                            <DB9>
30 PRINT "CC{5SPACE}MMMMMM EEEE{3SPACE}SSSSSS SSS
   SSS -------                                       <3K0>
40 PRINT "CC{5SPACE}MM{2SPACE}MM EE{9SPACE}SS{5SP
   ACE}SS                                            <MG3>
50 PRINT "CCCCCC MM{2SPACE}MM EEEEEE SSSSSS SSSSS
   S                                                 <O4R>
60 PRINT                                             <U7U>
70 PRINT "{7SPACE}TTTTTT OOOOOO UU{2SPACE}UU RRRR
   RR MM{2SPACE}MM AAAAAA MM{4SPACE}MM EEEEEE NN{
   2SPACE}NN TTTTTT                                  <B8S>
80 PRINT "{9SPACE}TT{3SPACE}OO{2SPACE}OO UU{2SPAC
   E}UU RR{2SPACE}RR MMM MM AA{2SPACE}AA MMM{2SPA
   CE}MMM EE{5SPACE}NNN NN{3SPACE}TT                 <UR9>
90 PRINT "{9SPACE}TT{3SPACE}OO{2SPACE}OO UU{2SPAC
   E}UU RRRRRR MMMMMM AAAAAA MM MM MM EEEE{3SPACE
   }NNNNNN{3SPACE}TT                                 <VDK>
100 PRINT "{9SPACE}TT{3SPACE}OO{2SPACE}OO UU{2SPA
    CE}UU RR RR{2SPACE}MM MMM AA{2SPACE}AA MM{4SP
    ACE}MM EE{5SPACE}NN NNN{3SPACE}TT                <TPN>
110 PRINT "{9SPACE}TT{3SPACE}OOOOOO UUUUUU RR{2SP
    ACE}RR MM{2SPACE}MM AA{2SPACE}AA MM{4SPACE}MM
     EEEEEE NN{2SPACE}NN{3SPACE}TT                   <U47>
120 PRINT : PRINT                                    <4SS>
130 PRINT " (W) 1986, 1987{2SPACE}REINHARD ANDREA
    S SCHULZ                                         <H31>
140 PRINT "{17SPACE}LINDEMANNSTRAßE 66               <SJQ>
150 PRINT "{17SPACE}4600 DORTMUND 1                  <NHE>
160 PRINT                                            <D5U>
170 PRINT " VIELEN DANK DEM SCHACHCLUB SF DO-BRAC
    KEL 1930 UND HERRN ANDREAS KROIS FÜR DIE         <ABL>
180 PRINT " FREUNDLICHE UNTERSTÜTZUNG!               <U36>
190 PRINT                                            <LAU>
191 XD$="->ST": FOR II=1 TO 4: POKE DEC("0400")+I
    I,ASC(MID$(XD$,II,1)): NEXT                      <D71>
200 PRINT "{23SPACE}+++++++++++++++++++++++++++++++
    +                                                <3SS>
210 PRINT : PRINT "BITE TASTE"                       <TI8>
220 GET KEY XT$                                      <QEA>
221 RUN "CH-VERWALTUNG"                              <4ID>
1000 RESUME                                          <901>
```

Listing 2. »LADER« lädt das Hauptprogramm nach. Bitte mit dem Checksummer 128 eingeben.

```
Name : ch.sys1            3800 4600
-----------------------------------
3800 : 0a 01 02 10 2b 3d 6a 7b  4b
3808 : 9a 9d a4 a5 3a 40 43 44  8a
3810 : 46 47 49 01 01 01 01 01  8a
3818 : 01 01 01 01 01 01 01 01  18
3820 : 01 3e 09 95 ff 9d a0 00  a5
3828 : a3 02 80 c0 e0 13 20 59  de
3830 : 3a a9 00 8d 00 08 8d 01  69
3838 : 08 8d 01 09 8d 00 09 8d  80
3840 : 02 09 ae 00 33 bd 00 33  fa
3848 : c9 ff d0 03 4c e1 38 a2  9f
3850 : 01 20 12 3c b0 03 e8 d0  d6
3858 : f8 20 58 3e 8d 21 38 8d  1c
3860 : 22 38 20 2a 3c ad 29 38  32
3868 : ac 26 38 c9 02 b0 19 ec  52
3870 : 28 38 d0 24 ad 00 09 f0  4e
3878 : 5d ad 00 08 cd 01 08 d0  54
3880 : 4d a9 00 8d 00 09 f0 4e  fc
3888 : 20 75 3c 90 0b 20 7d 39  9e
3890 : 90 03 4c b5 39 4c 56 38  2b
3898 : ad 00 08 f0 31 cd 02 09  01
38a0 : f0 2c 20 ad 3d ad 21 38  9a
38a8 : cd 23 38 d0 21 cd 24 38  b0
38b0 : d0 1c ad 00 09 f0 08 ad  8d
38b8 : 00 08 cd 01 09 d0 03 20  b3
38c0 : d0 3d 20 03 3e 20 2a 3c  9d
38c8 : 20 7a 3c 4c 8b 38 ad 00  ef
38d0 : 09 f0 03 20 29 3e ad 22  96
38d8 : 38 cd 00 38 b0 03 4c 2b  a9
38e0 : 39 ad 02 09 8d 00 08 8d  a6
38e8 : 01 08 ae 00 33 20 12 3c  8e
38f0 : b0 03 ca d0 f8 8a a8 88  86
38f8 : d0 1a ad 00 08 d0 06 a9  b3
3900 : 01 8d 20 38 60 20 03 3e  67
3908 : ad 00 08 8d 02 09 8d 01  0a
3910 : 08 88 f0 f1 20 b0 3c 90  70
3918 : de 20 7d 39 ad 00 08 8d  a3
3920 : 02 09 4c 4f 38 20 03 3e  b1
3928 : c8 d0 0b ac 22 38 c8 8c  d1
3930 : 22 38 b9 00 38 a8 20 95  51
3938 : 3c 90 0b 20 7d 39 b0 03  ee
3940 : 4c 56 38 4c b5 39 88 ad  f2
3948 : 00 08 cd 02 09 f0 2b 20  05
3950 : ad 3d ad 23 38 cd 24 38  5f
3958 : f0 20 cd 21 38 90 1b ad  c0
3960 : 24 38 cd 22 38 d0 13 ad  0a
3968 : 00 09 f0 08 ad 00 08 cd  c1
3970 : 01 09 d0 03 20 d0 3d 4c  a0
3978 : 25 39 4c ce 38 8e 2b 38  3c
3980 : 8c 2a 38 a2 02 ac 00 08  19
3988 : bd 29 38 99 02 08 c8 ca  34
3990 : d0 f6 8c 00 08 cc 01 08  f9
3998 : 90 0a f0 08 8c 01 08 a9  af
39a0 : 00 8d 00 09 cc 00 33 90  43
```

Listing 3. Geben Sie »CH.SYS1« bitte mit dem MSE (Seite 158) im C 64-Modus ein.

```
39a8 : 07 a9 00 8d 20 38 38 60  9b
39b0 : ae 2b 38 18 60 a5 a0 29  0d
39b8 : 01 18 69 01 8d 21 38 ae  60
39c0 : 00 08 d0 03 4c 9c 3f ca  95
39c8 : bc 02 08 b9 00 33 8d 2b  e5
39d0 : 38 ca bc 02 08 b9 00 33  91
39d8 : 8d 2a 38 8e 00 08 ad 2a  a6
39e0 : 38 c9 ff f0 3b ad 2b 38  59
39e8 : c9 ff f0 42 ae 00 10 ca  f6
39f0 : d0 0c ad 00 08 29 02 f0  e5
39f8 : 1c ad 21 38 d0 1c ae 2b  39
3a00 : 38 20 25 3d c9 04 b0 10  d9
3a08 : 8d 21 38 ae 2a 38 20 25  39
3a10 : 3d c9 00 d0 05 ad 21 38  ff
3a18 : 49 03 29 03 c9 02 f0 0e  1a
3a20 : ad 2a 38 48 ad 2b 38 8d  2a
3a28 : 2a 38 68 8d 2b 38 ae 00  69
3a30 : 30 ad 2b 38 9d 01 31 c9  43
3a38 : ff f0 07 ee 00 31 a9 00  7f
3a40 : f0 05 a9 08 ee 00 32 9d  11
3a48 : 01 32 ad 2a 38 9d 01 30  e8
3a50 : ee 00 30 ee 00 31 4c bf  62
3a58 : 39 20 66 40 20 e1 3b a2  86
3a60 : 00 a9 01 ac 00 30 f0 0d  6a
3a68 : d9 00 30 f0 29 d9 00 31  2f
3a70 : f0 24 88 d0 f3 e8 9d 00  ab
3a78 : 33 a8 b9 00 34 18 79 00  58
3a80 : 35 99 00 34 a9 00 99 00  09
3a88 : 36 99 00 06 99 00 37 99  f5
3a90 : 00 07 99 00 35 98 a8 c8  c7
3a98 : 98 cc 00 33 90 c5 f0 c3  7f
3aa0 : 8a 29 01 f0 14 e8 a9 ff  4c
3aa8 : 9d 00 33 a9 00 8d ff 34  1c
3ab0 : 8d ff 36 8d ff 37 8d ff  6c
3ab8 : 35 8e 00 33 20 47 3b a9  17
3ac0 : 01 8d 00 38 a2 02 8a 48  84
3ac8 : bd ff 32 a8 bd 00 33 aa  24
3ad0 : bd 00 34 d9 00 34 f0 0b  51
3ad8 : ee 00 38 68 48 ac 00 38  3c
3ae0 : 99 00 38 68 aa e8 ec 00  3a
3ae8 : 33 d0 db bd 00 33 c9 ff  f3
3af0 : d0 10 ee 00 38 ac 00 38  dd
3af8 : 8a 99 00 38 e8 8a 99 01  a1
3b00 : 38 60 8a 48 bc ff 32 b9  1c
3b08 : 00 33 aa bd 00 34 d9 00  0d
3b10 : 34 f0 04 68 aa d0 db 68  3c
3b18 : aa ac 00 38 d0 de 48 8a  5a
3b20 : 0a 0a 0a 0a 8d 2c 38 8a  23
3b28 : 4a 4a 4a 4a 18 69 10 8d  9b
3b30 : 2d 38 68 60 ae 00 33 8a  6c
3b38 : 9d 00 33 ca d0 f9 20 66  26
3b40 : 40 20 f4 40 20 e1 3b a2  19
3b48 : 01 8e 2a 38 8a a8 c8 8c  4c
3b50 : 2b 38 bd 00 33 aa b9 00  76
3b58 : 33 a8 bd 00 34 d9 00 34  c9
3b60 : d0 46 bd 00 35 d9 00 35  4f
3b68 : d0 3e bd 00 06 d9 00 06  02
3b70 : d0 36 bd 00 36 d9 00 36  69
3b78 : d0 2e bd 00 07 d9 00 07  1c
3b80 : d0 26 bd 00 37 d9 00 37  83
3b88 : d0 1e bd 00 0b d9 00 0b  6c
3b90 : d0 16 bd 00 0a d9 00 0a  5e
3b98 : d0 0e b9 00 05 dd 00 05  27
3ba0 : d0 06 b9 00 04 dd 00 04  19
3ba8 : b0 1c ac 2b 38 b9 00 33  af
3bb0 : 48 b9 ff 32 99 00 33 88  92
3bb8 : cc 2a 38 d0 f4 68 99 00  ba
3bc0 : 33 ae 2a 38 d0 88 ae 2a  3c
3bc8 : 38 ac 2b 38 c8 cc 00 33  82
3bd0 : f0 02 b0 03 4c 4f 3b e8  4c
3bd8 : ec 00 33 b0 03 4c 49 3b  d5
3be0 : 60 ae 00 33 20 1e 3b ad  39
3be8 : 2c 38 8d 02 3c ad 2d 38  2a
3bf0 : 18 69 10 8d 03 3c a9 00  2b
3bf8 : 9d 00 35 ac 00 10 88 f0  fd
3c00 : 0d b9 10 20 29 04 f0 03  6e
3c08 : fe 00 35 88 d0 f3 ca d0  de
3c10 : d3 60 48 8a ae 00 08 f0  64
3c18 : 08 dd 01 08 f0 08 ca d0  6c
3c20 : f8 aa 68 38 60 68 aa 68  53
3c28 : 18 60 98 48 8a 48 8d 2a  15
3c30 : 38 ae 21 38 bc 00 38 8c  d4
3c38 : 25 38 e8 bc 00 38 8c 27  8d
3c40 : 38 98 88 8c 28 38 38 ed  79
3c48 : 25 38 4a 18 6d 25 38 cd  9b
3c50 : 2a 38 f0 02 b0 03 ad 28  3d
3c58 : 38 8d 26 38 ae 25 38 a0  1e
3c60 : 00 20 12 3c 90 01 c8 e8  82
3c68 : ec 27 38 90 f4 8c 29 38  d1
3c70 : 68 aa 68 a8 60 20 b0 3c  9f
3c78 : b0 1a 8e 2a 38 cc 26 38  11
3c80 : 90 09 c8 cc 27 38 90 ed  b3
3c88 : ac 26 38 88 cc 2a 38 f0  47
3c90 : 02 b0 e2 18 60 98 48 ac  eb
3c98 : 22 38 b9 01 38 8d 27 38  62
3ca0 : 68 a8 cc 27 38 b0 ec 20  71
3ca8 : b0 3c 90 01 60 c8 d0 f2  30
3cb0 : 8a 48 98 48 aa 20 12 3c  fa
3cb8 : b0 0c 68 a8 68 aa 18 60  9a
3cc0 : 68 a8 68 aa 38 60 ad 00  29
3cc8 : 10 c9 02 90 f3 68 a8 68  45
3cd0 : aa 48 98 48 bd 00 33 aa  cb
3cd8 : b9 00 33 a8 20 1e 3b ad  ae
3ce0 : 2c 38 8d f2 3c ad 2d 38  40
3ce8 : 8d f3 3c 98 48 ac 00 10  9b
3cf0 : 88 d9 f0 18 d0 04 68 4c  0b
3cf8 : ba 3c 88 d0 f4 68 a8 c0  c3
3d00 : ff f0 0b e0 ff f0 07 20  3a
3d08 : 25 3d c9 04 b0 03 4c c0  95
3d10 : 3c 8d 2a 38 98 aa 20 25  4e
3d18 : 3d c9 04 90 f1 4d 2a 38  f0
3d20 : d0 ec 4c ba 3c 98 48 8a  8f
3d28 : 48 20 1e 3b ad 2c 38 8d  a8
3d30 : 4d 3d ad 2d 38 18 69 10  37
3d38 : 8d 4e 3d a9 ff 8d fe 30  3a
3d40 : 8d ff 30 a2 00 8e fd 30  fa
3d48 : ac 00 10 88 b9 e0 23 10  59
3d50 : 04 29 40 48 e8 88 d0 f4  02
3d58 : a0 00 e0 00 f0 3c 68 f0  a5
3d60 : 04 88 4c 66 3d c8 e0 03  2c
3d68 : b0 03 9d fd 30 ca d0 ea  33
3d70 : ae 00 10 e0 03 90 0c cd  bf
3d78 : ff 30 d0 07 a2 04 8e fd  25
3d80 : 30 d0 17 98 c9 00 f0 12  76
3d88 : 30 07 a2 00 8e fe 30 f0  68
3d90 : 05 a2 40 8e fe 30 29 01  e0
3d98 : f0 e2 68 aa 68 a8 ad fe  e9
3da0 : 30 f0 04 a9 02 d0 02 a9  81
3da8 : 01 0d fd 30 60 48 8a 48  b8
3db0 : 98 48 ac 00 08 be 01 08  22
3db8 : 20 58 3e 8d 24 38 ac 00  fc
3dc0 : 08 be 00 08 20 58 3e 8d  01
3dc8 : 23 38 68 a8 68 aa 68 60  75
3dd0 : ad 00 09 d0 0b 8c 04 09  11
3dd8 : 8e 05 09 a9 02 8d 00 09  ff
3de0 : 20 03 3e 8a 48 98 48 a2  12
3de8 : 02 ac 00 09 68 99 04 09  d7
3df0 : c8 ca d0 f8 8c 00 09 ac  b7
3df8 : 00 08 8c 01 09 c8 c8 8c  53
3e00 : 00 08 60 ac 00 08 88 88  25
3e08 : 8c 00 08 b9 02 08 48 aa  a4
3e10 : 20 58 3e 8d 21 38 ac 00  24
3e18 : 08 b9 03 08 48 aa 20 58  ca
3e20 : 3e 8d 22 38 68 a8 68 aa  77
3e28 : 60 ae 01 09 ac 00 09 88  41
3e30 : 88 b9 05 09 9d 02 08 e8  d3
3e38 : 88 d0 f6 8e 00 08 8c 00  2a
3e40 : 09 ad 04 09 48 aa 20 58  4d
3e48 : 3e 8d 22 38 ae 05 09 20  54
3e50 : 58 3e 8d 21 38 68 a8 60  79
3e58 : 8a ac 00 38 d9 00 38 b0  1f
3e60 : 03 88 d0 f8 98 60 a9 10  4e
3e68 : 8d 82 3e a9 30 8d 8b 3e  15
3e70 : d0 0a a9 30 8d 82 3e a9  ef
3e78 : 33 8d 8b 3e a9 00 a2 00  42
3e80 : 9d 00 33 e8 d0 fa ae 82  ac
3e88 : 3e e8 e0 33 8e 82 3e d0  70
3e90 : ed 60 ad 00 10 a2 00 a0  70
3e98 : 10 8c a1 3e 8d a0 3e 8e  12
3ea0 : f5 2f 18 69 10 90 f5 c8  4f
3ea8 : c0 30 90 ed ae 00 30 d0  af
3eb0 : 01 60 bd 00 30 8d 21 38  b5
3eb8 : bd 00 31 8d 22 38 bd 00  4e
3ec0 : 32 a8 c0 08 90 15 f0 08  fd
3ec8 : a9 10 8d 2a 38 4c 50 3f  c8
3ed0 : a9 1a 8d 2a 38 8d 2b 38  3c
3ed8 : 4c 50 3f a9 90 8d 2a 38  e0
3ee0 : a9 d0 8d 2b 38 88 10 13  e8
3ee8 : a9 ef 48 2d 2a 38 8d 2a  30
3ef0 : 38 68 2d 2b 38 8d 2b 38  1a
3ef8 : 4c 50 3f d0 09 ee 2a 38  77
3f00 : ee 2a 38 4c 50 3f 88 d0  5e
3f08 : 09 ee 2b 38 ee 2b 38 4c  1c
3f10 : 50 3f 88 d0 09 ee 2a 38  5d
3f18 : ee 2b 38 4c 50 3f 88 d0  f6
3f20 : 13 a9 04 48 0d 2a 38 8d  30
3f28 : 2a 38 68 0d 2b 38 8d 2b  2b
3f30 : 38 4c 50 3f a9 08 48 0d  a0
3f38 : 2a 38 8d 2a 38 68 0d 2b  78
3f40 : 38 8d 2b 38 88 d0 03 4c  c4
3f48 : fd 3e 88 d0 03 4c 09 3f  d6
3f50 : 8a 48 ae 22 38 f0 09 ad  79
3f58 : 21 38 ac 2b 38 20 75 3f  ff
3f60 : ae 21 38 ad 22 38 ac 2a  4e
3f68 : 38 20 75 3f 68 aa ca f0  de
3f70 : 03 4c b2 3e 60 48 98 48  49
3f78 : 20 1e 3b ad 2c 38 8d 95  12
3f80 : 3f 8d 99 3f ad 2d 38 8d  14
3f88 : 9a 3f 18 69 10 8d 96 3f  3b
3f90 : ac 00 10 68 99 e0 23 68  4b
3f98 : 99 e0 13 60 a2 01 a9 00  4b
3fa0 : 8d 00 34 8d ff 34 8e 2a  1c
3fa8 : 38 ac 2a 38 c8 8c 2b 38  d6
3fb0 : bd 00 30 8d 21 38 bd 00  f6
3fb8 : 31 8d 22 38 b9 00 30 8d  b7
3fc0 : 23 38 b9 00 31 8d 24 38  ee
3fc8 : ae 21 38 ac 22 38 bd 00  85
3fd0 : 34 d9 00 34 90 07 d0 0b  12
3fd8 : cc 21 38 b0 06 8c 21 38  13
3fe0 : b9 00 34 8d 27 38 ae 23  8d
3fe8 : 38 ac 24 38 bd 00 34 d9  e7
3ff0 : 00 34 90 07 d0 0b cc 23  ee
3ff8 : 38 b0 06 b9 00 34 8c 23  5b
4000 : 38 cd 27 38 d0 08 ad 21  36
4008 : 38 cd 23 38 f0 02 b0 1d  13
4010 : ac 2b 38 ae 2a 38 c8 cc  57
4018 : 00 30 f0 02 b0 03 4c ad  5c
4020 : 3f e8 ec 00 30 b0 03 4c  3c
4028 : a6 3f 4c c4 41 ac 2b 38  b0
4030 : b9 00 30 48 b9 00 31 48  ef
4038 : b9 00 32 48 b9 ff 2f 99  12
4040 : 00 30 b9 ff 30 99 00 31  f9
4048 : b9 ff 31 99 00 32 88 cc  ce
4050 : 2a 38 d0 e8 68 99 00 32  9f
4058 : 68 99 00 31 68 99 00 30  67
4060 : ae 2a 38 4c ac 3f a9 00  26
4068 : 8d ff 35 8d 00 35 ae 00  58
4070 : 33 8a 48 a9 00 9d 00 34  85
4078 : 20 1e 3b ad 2c 38 8d a1  2a
4080 : 40 8d be 40 ad 2d 38 8d  7f
4088 : bf 40 18 69 10 8d a2 40  13
4090 : 20 99 40 68 aa ca d0 d9  92
4098 : 60 ac 00 10 88 d0 01 60  24
40a0 : b9 10 20 29 03 18 7d 00  76
40a8 : 34 9d 00 34 88 d0 f1 9d  43
```

```
40b0 : 00 35 ac 00 10 88 f0 e7  4f
40b8 : a9 00 8d 2a 38 b9 10 10  bc
40c0 : d0 03 ee 2a 38 88 d0 f5  0a
40c8 : ad 2a 38 f0 d2 ad 00 10  71
40d0 : 4a 18 7d 00 35 a0 ff c8  70
40d8 : 38 ed 00 10 10 f9 98 ac  96
40e0 : 2a 38 8d 2a 38 bd 00 35  ab
40e8 : 18 6d 2a 38 88 d0 f9 9d  7b
40f0 : 00 35 60 60 ae 00 33 8a  7c
40f8 : 48 20 1e 3b ad 2c 38 8d  78
4100 : 2a 41 8d 53 41 8d 5c 41  0d
4108 : 8d 6e 41 8d 77 41 ad 2d  61
4110 : 38 8d 2b 41 8d 54 41 8d  9d
4118 : 5d 41 8d 6f 41 8d 78 41  4c
4120 : a9 00 8d 40 41 ac 00 10  ce
4128 : 88 be 10 10 18 7d 00 35  ed
4130 : 90 03 ee 40 41 88 d0 f1  85
4138 : a8 68 aa 98 9d 00 36 a9  d8
4140 : 00 9d 00 06 ac 00 10 88  ec
4148 : c0 03 b0 04 a9 00 f0 4c  2d
4150 : 8a 48 be 10 10 bd 00 35  09
4158 : 88 f0 0b be 10 10 dd 00  ec
4160 : 35 b0 ef 88 d0 f5 8d 2a  41
4168 : 38 ac 00 10 88 be 10 10  d7
4170 : bd 00 35 88 f0 0b be 10  0e
4178 : 10 dd 00 35 90 ef 88 d0  6a
4180 : f5 8d 2b 38 68 aa bd 00  e1
4188 : 06 9d 00 07 bd 00 36 38  63
4190 : ed 2a 38 b0 04 de 00 07  fc
4198 : 38 ed 2b 38 9d 00 37 b0  b1
41a0 : 03 de 00 07 ca f0 03 4c  cc
41a8 : f7 40 60 ad 20 38 ae 00  0c
41b0 : 30 f0 0d dd 00 30 f0 08  ad
41b8 : dd 00 31 f0 03 ca d0 f3  b1
41c0 : 8e 20 38 60 ad 00 30 8d  2f
41c8 : 2a 38 a2 01 bd 00 31 c9  0b
41d0 : ff f0 08 e8 ec 2a 38 90  89
41d8 : f3 b0 06 20 f5 41 4c d4  ed
41e0 : 41 a2 00 e8 ec 2a 38 90  b2
41e8 : 01 60 bd 00 31 d0 f4 20  36
41f0 : f5 41 4c e4 41 8a a8 b9  b4
41f8 : 00 30 48 b9 00 31 48 b9  78
4200 : 00 32 48 b9 01 30 99 00  5b
4208 : 30 b9 01 31 99 00 31 b9  4d
4210 : 01 32 99 00 32 c8 cc 2a  82
4218 : 38 90 e8 68 99 00 32 68  13
4220 : 99 00 31 68 99 00 30 88  7e
4228 : 8c 2a 38 60 14 01 20 59  60
4230 : 3a a9 00 8d 00 08 a2 01  bd
4238 : bd 00 33 cd 2c 42 f0 03  1a
4240 : e8 d0 f5 a0 01 b9 00 33  66
4248 : cd 2d 42 f0 03 c8 d0 f5  00
4250 : 20 b0 3c 90 04 a9 00 f0  59
4258 : 02 a9 01 8d 20 38 60 ae  c3
4260 : 2a 38 20 25 3d c9 04 90  a6
4268 : 06 29 03 c9 02 d0 04 a9  07
4270 : 00 f0 02 a9 01 8d 20 38  0c
4278 : 60 ae 00 33 8a 9d 00 33  92
4280 : ca d0 f9 20 66 40 20 e1  e1
4288 : 3b a9 00 ae 00 33 9d 00  7e
4290 : 36 9d 00 06 9d 00 37 9d  48
4298 : 00 07 ca d0 f1 4c 47 3b  fe
42a0 : 20 72 3e 20 66 40 20 e1  3a
42a8 : 3b ae 00 33 bd 00 34 18  7e
42b0 : 7d 00 35 9d 00 34 a9 00  77
42b8 : 9d 00 35 9d 00 36 9d 00  7f
42c0 : 37 9d 00 07 9d 00 06 ca  2e
42c8 : d0 e2 ae 00 33 8a ac 00  ef
42d0 : 30 f0 0d d9 00 30 f0 0b  52
42d8 : d9 00 31 f0 06 88 d0 f3  ec
42e0 : 20 e9 42 ca d0 e7 4c 9c  95
42e8 : 3f 8d 2a 38 48 20 1e 3b  f4
42f0 : ad 2c 38 8d 0f 43 8d 09  c6
42f8 : 43 ad 2d 38 8d 10 43 18  fb
4300 : 69 10 8d 0a 43 ac 00 10  d0
4308 : b9 40 21 8d 21 38 be 40  2b
4310 : 11 8e 2b 38 e0 ff d0 05  96
4318 : a9 08 4c b5 43 e0 00 d0  6c
4320 : 0f ad 21 38 29 10 d0 03  b2
4328 : 68 aa 60 a9 09 4c b5 43  83
4330 : ee 00 31 20 1e 3b ad 2c  3a
4338 : 38 8d 46 43 ad 2d 38 18  86
4340 : 69 10 8d 47 43 b9 10 20  80
4348 : 8d 22 38 ad 21 38 29 40  a3
4350 : f0 1c ad 2a 38 48 ad 2b  d2
4358 : 38 8d 2a 38 68 8d 2b 38  f9
4360 : ad 21 38 48 ad 22 38 8d  9d
4368 : 21 38 68 8d 22 38 ad 21  4e
4370 : 38 29 0f c9 04 d0 03 4c  a5
4378 : b5 43 c9 01 d0 05 a9 03  43
4380 : 4c b5 43 c9 02 d0 05 a9  bf
4388 : 01 4c b5 43 c9 0a d0 05  bf
4390 : a9 05 4c b5 43 c9 08 d0  ca
4398 : 13 ad 22 38 29 0f c9 08  54
43a0 : f0 05 a9 06 4c b5 43 a9  11
43a8 : 07 4c b5 43 ad 22 38 29  ca
43b0 : 0f f0 05 a9 02 ee 00 32  aa
43b8 : ee 00 30 ee 00 31 ac 00  cc
43c0 : 30 99 00 32 ad 2a 38 99  43
43c8 : 00 30 ad 2b 38 99 00 31  64
43d0 : 68 aa 60 a9 04 ae 00 30  f1
43d8 : f0 0c dd 00 32 f0 07 ca  a2
43e0 : d0 f8 a9 01 d0 02 a9 00  7b
43e8 : 8d 20 38 60 ae 00 30 f0  2d
43f0 : 19 bd 00 31 f0 11 c9 ff  cd
43f8 : f0 0d cd 20 38 f0 0d bd  a1
4400 : 00 30 cd 20 38 f0 05 ca  44
4408 : d0 e7 a2 00 8e 20 38 60  00
4410 : 32 c8 cc ff ff ff ff ff  d9
4418 : ff ff ff ff ff ff ff ff  17
4420 : ae 00 30 bd 00 31 c9 ff  43
4428 : f0 06 ca d0 f6 4c 2e 38  e3
4430 : ad 20 38 29 01 f0 f6 a9  e7
4438 : 01 8d 20 38 60 a9 00 8d  7d
4440 : 20 38 ad 00 30 cd 00 32  be
4448 : d0 07 20 d3 43 f0 02 60  a3
4450 : 00 a2 12 ad 21 38 f0 08  83
4458 : ca d0 0f ae 00 30 d0 0a  fd
4460 : e8 ec 00 30 90 04 f0 02  b5
4468 : a2 01 bd 00 32 f0 04 c9  49
4470 : 04 d0 e0 8e 52 44 60 ae  0c
4478 : 00 30 bd 00 32 f0 0c c9  6e
4480 : 08 b0 08 a9 00 9d 00 32  69
4488 : ce 00 32 ca d0 ec 60 ff  32
4490 : a0 00 a2 01 bd 00 32 f0  80
4498 : 04 c9 04 d0 05 c8 8a 99  90
44a0 : 00 08 e8 ec 00 30 90 ec  1a
44a8 : f0 ea 8c 00 08 60 ff ff  b4
44b0 : ff ff ff ff ff ff ff ff  af
44b8 : ff ff ff ff ff ff ff ff  b7
44c0 : 00 00 00 00 00 00 00 00  c1
44c8 : 00 00 00 00 00 00 00 00  c9
44d0 : 00 00 00 00 00 00 00 00  d1
44d8 : 00 00 00 00 00 00 00 00  d9
44e0 : 00 00 00 00 00 00 00 00  e1
44e8 : 00 00 00 00 00 00 00 00  e9
44f0 : 00 00 00 00 00 00 00 00  f1
44f8 : 00 00 00 00 00 00 00 00  f9
4500 : ff ff ff ff ff ff ff ff  ff
4508 : ff ff ff ff ff ff ff ff  07
4510 : ff ff ff ff ff ff ff ff  0f
4518 : ff ff ff ff ff ff ff ff  17
4520 : ff ff ff ff ff ff ff ff  1f
4528 : ff ff ff ff ff ff ff ff  27
4530 : ff ff ff ff ff ff ff ff  2f
4538 : ff ff ff ff ff ff ff ff  37
4540 : 00 00 00 00 00 00 00 00  41
4548 : 00 00 00 00 00 00 00 00  49
4550 : 00 00 00 00 00 00 00 00  51
4558 : 00 00 00 00 00 00 00 00  59
4560 : 00 00 00 00 00 00 00 00  61
4568 : 00 00 00 00 00 00 00 00  69
4570 : 00 00 00 00 00 00 00 00  71
4578 : 00 00 00 00 00 00 00 00  79
4580 : ff ff ff ff ff ff ff ff  7f
4588 : ff ff ff ff ff ff ff ff  87
4590 : ff ff ff ff ff ff ff ff  8f
4598 : ff ff ff ff ff ff ff ff  97
45a0 : ff ff ff ff ff ff ff ff  9f
45a8 : ff ff ff ff ff ff ff ff  a7
45b0 : ff ff ff ff ff ff ff ff  af
45b8 : ff ff ff ff ff ff ff ff  b7
45c0 : 00 00 00 00 00 00 00 00  c1
45c8 : 00 00 00 00 00 00 00 00  c9
45d0 : 00 00 00 00 00 00 00 00  d1
45d8 : 00 00 00 00 00 00 00 00  d9
45e0 : 00 00 00 00 00 00 00 00  e1
45e8 : 00 00 00 00 00 00 00 00  e9
45f0 : 00 00 00 00 00 00 00 00  f1
45f8 : 00 00 00 00 00 00 00 00  f9
```

Listing 3. (Schluß)

```
Name : centronics            0c00 0e05
--------------------------------------
0c00 : 78 a9 ff 8d 03 dd ad 02  d8
0c08 : dd 09 04 8d 02 dd a9 24  1b
0c10 : 8d 20 03 a9 0c 8d 21 03  5b
0c18 : a9 3f 8d 26 03 a9 0c 8d  52
0c20 : 27 03 58 60 20 02 f2 f0  ab
0c28 : 03 4c 82 f6 20 12 f2 a5  7a
0c30 : ba c9 04 f0 07 c9 05 f0  a3
0c38 : 03 4c 56 f1 4c 69 f1 48  9d
0c40 : a5 9a c9 04 d0 07 68 20  4c
0c48 : 81 0c 4c 59 0c c9 05 d0  d2
0c50 : 04 68 4c 59 0c 68 4c 79  ef
0c58 : ef c9 22 d0 08 a5 11 49  53
0c60 : ff 85 11 a9 22 8d 01 dd  ea
0c68 : ad 0d dd ad 00 dd 29 fb  54
0c70 : 8d 00 dd 09 04 8d 00 dd  fe
0c78 : a9 10 2c 0d dd f0 fb 18  5c
0c80 : 60 c9 23 d0 03 4c 3f 0d  51
0c88 : c9 0d d0 08 20 59 0c a9  5d
0c90 : 0a 4c 59 0c c9 40 d0 03  80
0c98 : 4c 32 0d c9 41 90 07 c9  c2
0ca0 : 5b b0 06 18 69 20 4c 59  53
0ca8 : 0c c9 60 b0 03 4c 3f 0d  71
0cb0 : c9 ac 90 f2 d0 05 a9 7b  25
0cb8 : 4c 4c 0d c9 ad d0 05 a9  70
0cc0 : 23 4c 59 0d c9 ae d0 05  61
0cc8 : a9 7d 4c 4c 0d c9 b0 d0  50
0cd0 : 05 a9 40 4c 3f 0d c9 b2  2c
0cd8 : d0 05 a9 40 4c 4c 0d c9  8c
0ce0 : b4 d0 05 a9 61 4c 70 0d  c7
0ce8 : c9 b5 d0 05 a9 65 4c 70  38
0cf0 : 0d c9 b6 d0 05 a9 69 4c  85
0cf8 : 70 0d c9 b7 d0 05 a9 6f  13
0d00 : 4c 70 0d c9 b8 d0 05 a9  7a
0d08 : 75 4c 70 0d c9 bb 90 0a  32
0d10 : c9 bf b0 09 38 e9 40 4c  73
0d18 : 32 0d 4c 59 0c c9 c1 90  46
0d20 : f9 c9 db b0 06 38 e9 80  d6
0d28 : 4c 59 0c c9 de b0 eb 38  f1
0d30 : e9 80 48 20 66 0d a9 02  e9
0d38 : 20 59 0c 68 4c 59 0c 48  65
0d40 : 20 66 0d a9 00 20 59 0c  8a
0d48 : 68 4c 59 0c 48 20 66 0d  e7
0d50 : a9 01 20 59 0c 68 4c 59  95
0d58 : 0c 48 20 66 0d a9 03 20  c8
0d60 : 59 0c 68 4c 59 0c a9 1b  36
0d68 : 20 59 0c a9 52 4c 59 0c  72
0d70 : 20 59 0c a9 08 20 59 0c  74
```

Listing 4. »CENTRONICS« initialisiert eine Centronics-Schnittstelle im C 128. Bitte mit dem MSE eingeben.

```
0d78 : a9 5e 4c 59 0c 20 00 0c  68
0d80 : a9 1b 8d 01 dd ad 0d dd  76
0d88 : ad 00 dd 29 fb 8d 00 dd  ba
0d90 : 09 04 8d 00 dd a9 10 a2  af
0d98 : 00 a0 00 2c 0d dd d0 10  91
0da0 : c8 c0 ff d0 f6 e8 e0 ff  1c
0da8 : d0 ef a9 01 8d ba 0d 60  9e
0db0 : a9 00 8d ba 0d a9 40 4c  cc
0db8 : 59 0c 00 00 82 14 ea 84  95
0dc0 : 4a 20 79 00 20 13 b1 b0  3b
0dc8 : 09 a9 20 85 81 a9 07 4c  79
0dd0 : 7f 0a 20 73 00 90 05 20  a4
0dd8 : 13 b1 90 04 e8 4c d2 0d  bf
0de0 : c9 27 f0 f8 e8 86 45 a9  c3
0de8 : 36 85 01 a9 f9 a2 bf 86  17
0df0 : 60 85 5f e4 30 90 4a d0  da
0df8 : 04 c5 2f 90 44 a0 01 8d  25
0e00 : 03 dd ad 02 dd a9 12 4c  a9
```

Listing 4. (Schluß)

```
Name : screen              0b20 0b7a
------------------------------------
0b20 : a9 09 8d 32 0b a9 90 48  c8
0b28 : a2 08 a0 02 a9 12 20 6e  bf
0b30 : 0b a9 0e 8d 01 d6 a9 13  d9
0b38 : 20 6e 0b 68 48 8d 01 d6  02
0b40 : a9 1f 20 6e 0b c0 01 d0  ab
0b48 : 04 a9 8e d0 02 a9 8f 8d  a5
0b50 : 01 d6 a9 1e 20 6e 0b a9  df
0b58 : 4f 8d 01 d6 68 18 69 50  17
0b60 : 48 90 03 ee 32 0b 88 d0  ce
0b68 : c3 ca d0 be 68 60 8d 00  5c
0b70 : d6 2c 00 d6 10 fb 60 00  9a
0b78 : ef c9 0c d4 8d 13 d4 a9  02
```

Listing 5. »SCREEN« wird vom Hauptprogramm nachgeladen. Bitte mit dem MSE eingeben.

```
Name : spsort              1500 168d
------------------------------------
1500 : 20 26 15 a9 60 8d 5c 15  bc
1508 : a9 20 8d a5 15 4c 72 15  81
1510 : 20 26 15 a9 ea 8d 5c 15  75
1518 : 4c 31 15 20 26 15 a9 2c  50
1520 : 8d a5 15 4c 72 15 ae 00  d9
1528 : 14 8a 9d 00 14 ca d0 f9  b8
1530 : 60 a2 01 8e 7e 16 8a a8  08
1538 : c8 8c 7f 16 ae 7e 16 20  60
1540 : df 15 ae 7f 16 a9 80 20  36
1548 : e9 15 a2 00 bd 80 13 dd  4c
1550 : 00 13 d0 03 e8 10 f5 b0  b7
1558 : 03 20 c4 15 ea ae 7e 16  89
1560 : ac 7f 16 c8 cc 00 14 90  a9
1568 : d0 f0 ce e8 ec 00 14 90  c1
1570 : c2 60 a2 01 8e 7e 16 8a  75
1578 : a8 c8 8c 7f 16 bd 00 14  0f
1580 : aa b9 00 14 a8 bd 00 cc  9c
1588 : d9 00 cc d0 1d bd 00 cd  0a
1590 : d9 00 cd d0 15 b9 00 ce  b3
1598 : dd 00 ce d0 0d b9 00 cf  81
15a0 : dd 00 cf d0 05 2c 3c 15  58
15a8 : 90 05 b0 03 20 c4 15 ae  21
15b0 : 7e 16 ac 7f 16 c8 cc 00  2f
15b8 : 14 90 bf f0 bd e8 ec 00  f9
15c0 : 14 90 b1 60 ac 7f 16 b9  27
15c8 : 00 14 48 b9 ff 13 99 00  1b
15d0 : 14 88 cc 7e 16 d0 f4 68  b8
15d8 : 99 00 14 8c 7f 16 60 a9  85
15e0 : 00 a0 00 99 00 13 c8 d0  c1
15e8 : fa 8d 08 16 20 69 16 8d  2e
15f0 : fb 15 8c fc 15 a2 00 8e  bc
15f8 : 80 16 bd e6 df c9 0d f0  32
1600 : 18 20 1d 16 ae 80 16 9d  b5
1608 : 80 13 c0 00 f0 08 ee 08  5d
1610 : 16 98 a0 00 f0 f1 e8 10  fd
1618 : de 60 20 1d 16 a2 09 dd  28
1620 : 3b 16 f0 0b dd 56 16 f0  cf
1628 : 0e ca d0 f3 a0 00 60 bd  55
1630 : 44 16 bc 4d 16 d0 f7 bd  9b
1638 : 5f 16 d0 f0 bd bc be bb  29
1640 : db dc dd be be 55 4f 53  53
1648 : 41 c1 cf d5 53 53 45 45  88
1650 : 53 45 45 45 45 53 53 ae  d9
1658 : ac b4 b5 b6 b3 b2 b7 b8  c4
1660 : 45 45 41 45 49 55 41 4f  24
1668 : 55 bd 00 14 aa a9 00 a0  58
1670 : d0 ca f0 09 18 69 25 90  85
1678 : f8 c8 4c 71 16 60 00 00  7a
1680 : 78 a9 6b 8d 28 03 58 60  16
1688 : 78 a9 6e d0 f6 00 03 00  06
```

Listing 6. »SPSORT« sorgt für schnelles Sortieren. Bitte mit dem MSE eingeben.

Tips und Tricks für den Einstieg in Maschinensprache

Um jedem C128-Besitzer den Einstieg in Maschinensprache zu erleichtern, schildert hier ein Assembler-Profi seine Erfahrungen mit diesem Computer und hilft Ihnen mit vielen Tips.

Das »Bankswitching«, also das Umschalten zwischen mehreren sich überlagernden Speicherbereichen, ist bereits vom C64 her bekannt, bei dem auf diese Weise RAM-Bereiche unter dem ROM genutzt werden können. Beim C64 war es jedoch nicht unbedingt erforderlich, über Speicherkonfigurationen und deren Änderung Bescheid zu wissen. Viele Assemblerprogrammierer dürften sich – ebenso wie ich selbst – so weit wie möglich um diese etwas »ominöse« Geschichte herumgedrückt haben.

Die Programmierung des C128 in Maschinensprache ist leider ohne Kenntnis des Bankswitching kaum möglich. Wie Sie bereits wissen, verwendet der Basic-Interpreter Bank 0 für das Basic-Programm und Bank 1 für die Variablenspeicherung. Der Interpreter ist daher gezwungen, ständig zwischen diesen 64-KByte-RAM-Bänken hin- und herzuschalten.

Das gleiche Problem stellt sich für den Assemblerprogrammierer. Eine beliebte Anwendung von Assemblerprogrammen sind Unterroutinen, die mit dem Basic zusammenarbeiten sollen, zum Beispiel um besonders zeitkritische Programmteile zu beschleunigen, oder aber, um nicht vorhandene Basic-Befehle zu implementieren. Werden solche Routinen auf dem C64 geschrieben, kann sie der Programmierer problemlos in einen freien, vom Überschreiben durch Basic geschützten Speicherbereich legen (zum Beispiel $C000 bis $CFFF).

Der C128 erfordert auch bei kleineren Assembler-Routinen einen weitaus höheren Aufwand. So stellt einen bereits die Frage, in welchen Speicherbereich eine Routine gelegt werden soll, vor größere Probleme. Es stehen zwar 128 KByte zur Verfügung, jedoch kein größerer Speicherbereich, der vom Basic mit Sicherheit verschont bliebe. Bank 0 steht bis $FFFF für den Basic-Text, und Bank 1 ebenfalls bis $FFFF für Variablen zur Verfügung. Weiterhin muß in vielen Fällen eine Assembler-Routine auf den Basic-Text und auch auf die Variablen zugreifen, was ohne Umschaltung der Konfiguration nicht möglich ist. Zum Beispiel muß eine Sortierroutine Parameter aus dem Basic-Text lesen (Name des zu sortierenden Arrays), der eigentliche Sortiervorgang findet anschließend in Bank 1, der Variablenbank, statt.

Zusammen mit einem Programmierkollegen stand ich vor der Aufgabe, ein Basic-Programm vom C64 auf den C128 umzuschreiben. Nach einigen Fehlschlägen gelang

es uns, alle dabei benötigten Assemblerroutinen umzuwandeln.

Begonnen hatte alles mit einem C128, dem Handbuch und einem Exemplar des »C128 Intern«. Unser erster Schritt war ein Testprogramm, mit dem das Umschalten zwischen den Speicherbänken erprobt werden sollte. Das Programm sollte eine Speicherstelle in Bank 1 lesen. Wir gaben es mit dem integrierten Monitor in Bank 0 ein, und legten es - wie vom C64 gewohnt - nach $C000. Nach dem Aufruf sollte es durch Beschreiben des Konfigurationsregisters $FF00 auf Bank 1 umschalten, die gewünschte Speicherstelle in den Akku einlesen und mit einem BRK in den Monitor zurückkehren.

Dieses einfache Testprogramm stürzte jedoch ab. Der Grund für den Absturz war schnell gefunden: nach dem Beschreiben von $FF00 und damit dem Umschalten auf Bank 1 folgte der Befehl »LDA $1000«, der den Inhalt dieser Speicherstelle lesen sollte. Da unser Programm jedoch in Bank 0 lag, sägten wir uns mit dem Umschalten auf Bank 1 gewissermaßen »den eigenen Ast ab«, da unser Programm - und damit auch der Ladebefehl - nach dem Umschalten auf Bank 1 für den Prozessor nicht mehr existent war.

Dieser Mißerfolg machte uns immerhin klar, daß wir zu naiv an das Problem herangegangen waren. Nachdem wir einige Zeit damit verbrachten, das Handbuch und das C128 Intern zu studieren, war klar, daß es prinzipiell zwei Methoden gibt, auf verschiedene Speicherbänke zuzugreifen, ohne daß ein Programm unmittelbar nach dem Umschalten abstürzt:

Abstürze verhindern

1. Benutzung der Routinen FETCH, STASH und CMPARE:

Diese Routinen benutzt auch der Basic-Interpreter. Nach dem Einschalten des Computers ist der Bereich $0000 bis $03FF als gemeinsamer Bereich von Bank 0 und Bank 1 definiert. Unabhängig davon, welche Bank eingeschaltet ist, »sieht« der Prozessor immer die gleichen Programme beziehungsweise Daten in diesem Bereich. Dieses Konzept der gemeinsamen Speicherbereiche kann man sich durch die Vorstellung veranschaulichen, daß jede Änderung des Bereichs $0000 bis $03FF in der einen Bank sofort in die andere Bank kopiert wird, die Inhalte von Bank 0 und 1 in diesem Bereich daher in jedem Moment identisch sind. Dieses Konzept entspricht zwar nicht dem tatsächlichen physischen Ablauf, ist jedoch eine für die Praxis völlig ausreichende Vorstellung.

Die genannten Routinen werden nach dem Einschalten des Computers aus dem ROM in diesen gemeinsamen Bereich kopiert. Ihre Benutzung erfordert eine recht langwierige Parameterübergabe. Unter anderem wird die gewünschte Speicherkonfiguration übergeben, auf die zugegriffen werden soll. Nach der Parameterübergabe und dem Aufruf schaltet die jeweilige Routine die benötigte Konfiguration ein, führt in der ausgewählten Bank die gewünschte Operation aus (lesen, schreiben, vergleichen) und schaltet danach auf die alte Speicherkonfiguration zurück.

Der Einsprung in eine dieser Routinen findet in das Kernel statt. Danach wird jedoch in die Kopie, also in den gemeinsamen Speicherbereich verzweigt. Da das Umschalten in diesem gemeinsamen Bereich durchgeführt wird, findet kein Absturz statt, da die Routine für den Prozessor in jeder Bank existent ist. Diese Routinen sind daher das elementare »Verbindungsglied« zwischen Programmen und Daten, die in verschiedenen Bänken liegen.

2. Assembler-Routinen in einen gemeinsamen Speicherbereich legen:

Wie erwähnt, wird beim Einschalten des C128 ein bestimmter Speicherbereich als gemeinsam definiert. Über das Register $D506, das im I/O-Bereich liegt, kann der Programmierer jedoch selbst bestimmen, ob er gemeinsame Bereiche definieren will, wo und wie groß diese sein sollen.

Es existieren jedoch mehrere Begrenzungen, zum Beispiel dadurch, daß ein gemeinsamer Bereich an einem Ende des Speichers beginnen muß (entweder ab $0000 aufwärts oder ab $FFFF abwärts) und durch die maximale Größe eines solchen Bereichs von 32 KByte.

Der gemeinsame Speicherbereich

Die zweite Möglichkeit besteht somit darin, einen Speicherbereich als gemeinsam zu deklarieren, der groß genug ist, alle benötigten Assembler-Routinen aufzunehmen.

Da die Routinen FETCH etc. langwierige Parameterübergaben erfordern, ein Programm daher länger wird und zusätzliche Fehlerquellen entstehen, waren mein Kollege und ich uns einig, die zweite Möglichkeit zu verwenden. Die Frage war nun, welcher Speicherbereich als gemeinsam deklariert und für die Routinen verwendet werden sollte. Wir entschieden uns dafür, den oberen Speicherbereich bis $FFFF zu verwenden. Voraussetzung dafür war natürlich, den Zeiger auf das Ende des Basic-Textes herabzusetzen, um die Routinen vor dem Überschreiben zu schützen.

Zusätzlich war es erforderlich, die Zeiger auf den Anfang des Stringbereichs ebenfalls herabzusetzen. Der Stringstack beginnt ab $FFFF in Bank 1. Wenn nun zum Beispiel der Speicherbereich $F000 bis $FFFF als gemeinsamer Bereich definiert wird, führt jedes Anlegen eines Strings dazu, daß die Assembler-Routinen in dem entsprechenden Bereich von Bank 0 durch den String überschrieben würden.

Nach den frustrierenden Erfahrungen mit dem ersten Testprogramm versuchten wir nun, möglichst alle Konsequenzen unseres Plans zu überdenken. Dabei stellte sich ein neues Problem: Im gleichen Bereich befindet sich das Kernel, das zum Zugriff auf unsere Routinen ausgeschaltet werden mußte. Diese Routinen benötigten jedoch selbst mehrere Kernel-Routinen.

Zur Benutzung von ROM-Routinen trotz ausgeblendetem ROM stehen die Routinen JRSFAR und JMPFAR zur Verfügung. Bei der Benutzung dieser Routinen muß das Kernel-ROM eingeschaltet sein. Die Routinen selbst befinden sich zwar nicht im Kernel, verzweigen jedoch zu einer Kernel-Routine, was bei ausgeblendetem Kernel natürlich zu einem Absturz führen würde.

JSRFAR stellt daher leider für Programme, die sich unter dem Kernel befinden und dessen Routinen verwenden wollen, keine brauchbare Lösung dar. Denkbar ist jedoch ein Umweg, mit dem das Ziel gewissermaßen über mehrere Ecken angesteuert wird:

Zwei gemeinsame Bereiche werden definiert, einer am unteren und einer am oberen Ende des Speichers. In den unteren Bereich, der nicht vom ROM überlagert wird, wird eine kleine Routine gelegt, die den Aufruf einer vom Hauptprogramm benötigten Kernel-Routine übernimmt. Das Hauptprogramm übergibt alle Parameter (zum Beispiel die Adresse der Kernel-Routine) und ruft die Routine im unteren Bereich auf. Diese blendet das Kernel ein und dadurch zwangsläufig das Hauptprogramm aus. Nun wird JSRFAR aufgerufen. Nach Durchführung der jeweiligen Kernel-Routine wird mit einem RTS zu der Routine im unteren gemeinsamen Bereich zurückgekehrt. Diese blendet nun das Kernel wieder aus und das darunterliegende Hauptprogramm ein; anschließend erfolgt die Rückkehr in dieses Hauptprogramm.

Wie Sie sehen – und auch wir einsehen mußten – ist es ohne größeren Aufwand nicht möglich, ein Assemblerprogramm, das Kernel-Routinen verwendet, unter eben dieses Kernel zu legen. Die Umschaltung zwischen Kernel und darunterliegendem Programm führt zu sehr umständlichen Routinen.

Ich möchte Ihnen nun zwei kleine Initialisierungs-Routinen vorstellen, die die gewünschten Aufgaben erledigen, eine Basic- und eine Assembler-Routine:

1. Beginn des Basic-Textes und der Variablentabelle verschieben:

```
10 poke 46,dec("20"):poke 48,dec
("20")
20 poke dec("2000"),0
30 clr
40 new
```

Wenn Sie dieses Basic-Programm eingeben und starten, beginnt sowohl der Basic-Text als auch die Variablentabelle ab $2000.

2. Definition eines gemeinsamen Speicherbereiches:

```
a 00b00  ad  00 ff  lda $ff00
a 00b03  48         pha
a 00b04  a9  00     lda #$00
a 00b06  8d  00 ff  sta $ff00
a 00b09  ad  06 d5  lda $d506
a 00b0c  09  06     ora #$06
a 00b0e  8d  06 d5  sta $d506
a 00b11  68         pla
a 00b12  8d  00 ff  sta $ff00
a 00b15  60         rts
```

Die einzige von uns gefundene Möglichkeit, kleinere Assembler-Routinen auf dem C128 zu schreiben, ohne an derartigen Problemen zu scheitern, besteht darin, die Programme in einen Bereich zu legen, der nicht vom ROM – das eventuell benötigt wird – überlagert und als gemeinsamer Bereich von Bank 0 und Bank 1 deklariert wird.

Ein solcher Bereich, der für die meisten Assembler-Routinen ausreichend Platz bietet und unbenutzt ist, ist der Bereich von $1300 bis $17FF. Um Probleme mit dem Umschalten zwischen den Bänken zu vermeiden, wird ein gemeinsamer Bereich definiert, der von $0000 bis $1FFF reicht. Da der Basic-Text jedoch ab $1C00 beginnt, muß der Zeiger auf den Basic-Anfang auf $2000 verstellt werden, um Überschneidungen zwischen dem Basic-Text in Bank 0 und den Variablen in Bank 1 zu vermeiden.

Zusätzlich muß auch der Zeiger auf den Beginn der Variablentabelle verstellt werden – auch auf $2000 –, da diese

Bankswitching

ab $0800 beginnt und ebenfalls ab $0800, jedoch in Bank 0, der Bildschirmspeicher liegt. Überschneidungen zwischen Variablentabelle und Bildschirmspeicher führen sonst bei jeder Ausgabe auf dem Bildschirm zu Änderungen der Variablentabelle, also zu ihrer Zerstörung.

Dieses kleine Maschinenprogramm rettet zuerst die aktuelle Speicherkonfiguration, blendet nun den I/O-Bereich ein und definiert im RAM-Konfigurationsregister den Bereich $0000 bis $1FFF als gemeinsamen Bereich von Bank 0 und Bank 1. Vor dem Rücksprung aus der Routine wird die gerettete Speicherkonfiguration wiederhergestellt. Diese Routine liegt im Kassettenpuffer. Sollten Sie mit der Datasette arbeiten, legen Sie sie bitte nach $0D00 in den RS232-Eingabepuffer.

Wenn Sie beide Programme eingegeben und gespeichert haben, können Sie mit der Assemblerprogrammierung beginnen. Bis Sie selbst eine komfortablere Lösung

RAM Configuration Register

Bits	Bedeutung
7,6	RAM-Bank des VIC (Bit 7 für Erweiterung) x0 = RAM-Bank 0 x1 = RAM-Bank 1
5,4	Nicht benutzt (für Erweiterung auf 1 MByte)
3,2	Lage der gemeinsamen Speicherbereiche 00 = keine gemeinsamen Speicherbereiche 01 = Speicheranfang 10 = Speicherende 11 = Speicheranfang und -Ende
1,0	Größe des/der gemeinsamen Speicherbereiche 00 = 1 KByte 01 = 4 KByte 10 = 8 KByte 11 = 16 KByte

Tabelle 1. Das RAM-Konfigurations-Register ($D506) der MMU

gefunden haben, gehen Sie bei der Assemblerprogrammierung bitte wie folgt vor:

1. Basic-Programm laden und starten.
2. Maschinenprogramm laden und mit »SYS DEC ("0B00")« aufrufen.

Für Ihre Maschinenprogramme steht Ihnen nun der Bereich $1300 bis $17FF (ab $1800 beginnt der für die Funktionstastenbelegung benötigte Bereich) zur Verfügung. Eventuell können Sie auch weitere Bereiche nutzen, zum Beispiel $0C00 bis $0DFF, wenn Sie die RS232-Schnittstelle, oder den Bereich von $0B00 bis $0BFF, wenn Sie den Kassettenpuffer nicht benötigen.

Die Nutzung aller Bereiche unterhalb $2000 ist theoretisch möglich. Probleme durch das »Absägen des eigenen Astes« werden dank dem bis $1FFF reichenden, gemeinsamen Speicherbereich nicht auftreten.

Um wirklich problemlos in Assembler arbeiten zu können, benötigen Sie zwei weitere Routinen, die zwischen Bank 0 und Bank 1 umschalten. Diese Routinen wurden von mir in das Initialisierungs-Programm integriert.

Initialisierung + Umschaltroutinen:

```
a 00b00  ad  00 ff  lda $ff00
a 00b03  48         pha
a 00b04  a9  00     lda #$00
a 00b06  8d  00 ff  sta $ff00
a 00b09  ad  06 d5  lda $d506
a 00b0c  09  06     ora #$06
a 00b0e  8d  06 d5  sta $d506
a 00b11  68         pla
a 00b12  8d  00 ff  sta $ff00
a 00b15  60         rts
a 00b16  08         php
a 00b17  48         pha
a 00b18  a9  00     lda #$00
a 00b1a  8d  00 ff  sta $ff00
a 00b1d  68         pla
a 00b1e  28         plp
a 00b1f  60         rts
a 00b20  08         php
a 00b21  48         pha
a 00b22  a9  7f     lda #$7f
a 00b24  8d  00 ff  sta $ff00
a 00b27  68         pla
a 00b28  28         plp
a 00b29  60         rts
```

$0B16 ist die Einsprungadresse zum Umschalten auf Bank 0, $0B20 der Einsprung zum Umschalten auf Bank 1 (nur RAM), die wichtigere der beiden Routinen. Wichtiger, da Sie Bank 0 wohl nur zum Lesen von Parametern aus dem Basic-Text verwenden werden, wozu problemlos die entsprechenden Routinen des Basic-Interpreters verwendet werden können.

Assemblerprogrammierung

Im Laufe der Zeit werden sicherlich auch elegantere Lösungen der genannten Probleme entdeckt werden. Dieser Artikel sollte nur zeigen, worauf Sie bei der Assemblerprogrammierung des C128 achten müssen:

1. Probleme beim Umschalten zwischen den Speicherbänken.
2. Probleme mit Überschneidungen zwischen als gemeinsam definierten Speicherbereichen, zum Beispiel Bildschirm und Variablentabelle.
3. Probleme bei dem Zugriff auf Betriebssystem-Routinen, deren Lösung davon abhängt, in welchen Bereich die Programme gelegt werden, unter das ROM oder in einen reinen RAM-Bereich.

Wie Sie sehen, ist es bei der Programmierung des C128 in Assembler außerordentlich wichtig, alle Zugriffe, die im späteren Programm sowohl auf verschiedene RAM-Bänke als auch auf ROM-Routinen erfolgen sollen, von vornherein einzuplanen.

Zum Abschluß noch ein Rat: Sie werden sich vor weitere Probleme gestellt sehen, wenn Sie Programme, die mit einer bestimmten Art der Bildschirmdarstellung arbeiten, zum Beispiel dem 40-Zeichen-Modus, auf einen anderen Ausgabemodus umstellen wollen.

Der Grund ist der unterschiedliche Videocontroller. Für den 40-Zeichen-Modus ist der VIC, für den 80-Zeichen-Modus der VDC zuständig. Beide Controller arbeiten unterschiedlich. Während der VIC den direkten Zugriff auf das Video-RAM gestattet, ist beim VDC nur ein indirekter Zugriff möglich.

Beim C64 ist es üblich, daß Assembler-Routinen direkt – unter Umgehung der Betriebssystem-Routinen – in das Video-RAM schreiben beziehungsweise daraus lesen. Bei einer eventuellen Umstellung der Zeichendarstellung sind in solchen Routinen umfangreiche Änderungen nötig.

Ich empfehle Ihnen daher, ausschließlich die Betriebssystem-Routinen – zum Beispiel BSOUT – zu verwenden. Wenn diese standardisierten Schnittstellen verwendet werden, können Sie sicher sein, daß eine spätere Programmumstellung auf eine andere Art der Bildschirmdarstellung nicht einem Neuschreiben des Programms gleichkommt.

Daß mit BSOUT Zeichen auf dem Bildschirm ausgegeben werden können, dürfte allgemein bekannt sein. Woran jedoch nicht jeder Programmierer denkt, ist die Möglichkeit, ohne Umgehung des Betriebssystems auch Zeichen vom Bildschirm zu lesen, indem dieser als logische Datei eröffnet und die Eingabe auf diese Datei gelegt wird.

Zweifellos eine umständliche Methode, die jedoch bei späteren Programmänderungen viel Ärger ersparen kann.

(S. Baloui/kn)

Grafik richtig programmieren

Das Basic des C128 verfügt über wunderbare Befehle zur Nutzung der HiRes-Grafik. Wir zeigen Ihnen Anwendung und Programmierung dieser HiRes-Grafik.

Hier geht es um die hochauflösende und die Mehrfarbengrafik, die wir uns im Detail ansehen werden. 14 Befehle steuern im Basic 7.0 des C128 die Grafikmöglichkeiten im engeren Sinn. Zählt man noch die Sprite- und Shapeanweisungen hinzu, dann verfügen wir über 27 grafische Werkzeuge in Basic. Wir erklären Ihnen die Anwendung und Programmierung jedes einzelnen dieser 14 Befehle, die Ihnen das Arbeiten mit der hochauflösenden Grafik erleichtern. Hier gehen wir nicht auf die VDC-Änderungen des neuen C 128 D im Blechgehäuse ein. Dies wurden im Sonderheft 29 ausführlich besprochen.

GRAPHIC

Dieser Befehl dient – der Name sagt es schon – zum Umschalten in die verschiedenen Grafik- und Textmodi unseres Computers. Mittels

```
GRAPHIC A, B, C
```

können je nach den Kennungen A, B und C eine ganze Reihe von Optionen angewählt werden.

Die Kennung A aktiviert den Grafik-Modus:

A=0 Der Textbildschirm – und zwar derjenige mit 40 Zeichen Zeilenbreite – wird eingeschaltet

A=1 Die hochauflösende Grafik wird aktiviert

A=2 Ein »Splitscreen«, also ein Bildschirm, auf dem sowohl 40-Zeichen-Text als auch hochauflösende Grafik zulässig sind, kann damit eingeschaltet werden

A=3 Die Mehrfarbengrafik (auch Multicolorgrafik genannt) ist angewählt

A=4 Auch dieser Parameter erlaubt die Einrichtung eines Splitscreen. Diesmal erscheint zum 40-Zeichen-Text-Schirm noch die Mehrfarben-Grafik

A=5 Schaltet auf 80 Zeichen um.

Der Parameter B entscheidet darüber, ob der ausgewählte Bildschirm gelöscht wird oder nicht:

B=0 Es wird nicht gelöscht

B=1 Der Bildschirm (in den Grafik-Modi 1 bis 4 auch die

Bit-Map genannt) wird gelöscht.

Der Parameter C spielt lediglich in den Modi 2 und 4 eine Rolle (Splitscreen). Er bestimmt, an welcher Zeile der Textbildschirm eingeblendet werden soll. Die Splitscreens sind so eingerichtet, daß immer von dieser Zeile an abwärts Textbereich existiert. Voreingestellt - und daher muß C nicht immer angegeben werden - ist die Zeile 19.

Worin unterscheiden sich diese Grafik-Modi? Wenn Sie im C 64-Modus Grafik programmieren wollen, dann kommen Sie ohne einen sehr tiefen Einblick in die Speicherorganisation des Computers nicht sehr weit. Auch wenn uns das der C 128 durch sein leistungsfähigeres Basic erspart: Ein wenig sollten Sie dennoch wissen.

Die Grafik-Modi

40-Zeichen-Text: Zur Darstellung des 40-Zeichen-Bildschirms belegt der Computer zwei Speicherbereiche:

1024 bis 2023 Bildschirm-RAM und
55296 bis 56295 Bildschirm-Farben-RAM

In jeweils 1000 Speicherstellen (40 Spalten mal 25 Zeilen) befindet sich die Text-Information, die auf dem Monitor oder Fernseher zu sehen ist. Die Aufteilung dieser beiden Speicherbereiche finden Sie in Ihrem C 128-Handbuch. Wenn Sie dafür sorgen, daß in der linken oberen Ecke Ihres Bildschirms ein A steht, dann finden Sie durch

```
BANK 0:PRINT PEEK(1024)
```

die Zahl 1 in der Speicherstelle 1024. Das ist der Bildschirm-Code des Zeichens »A«. Umgekehrt können Sie

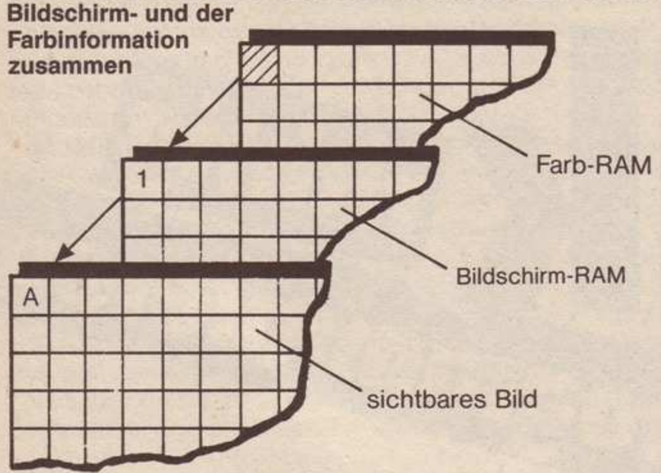


Bild 1. Ein Zeichen auf dem Bildschirm setzt sich aus der Bildschirm- und der Farbinformation zusammen

durch POKEn auch Zeichen in den Bildschirmspeicher schreiben:

```
BANK 0:POKE 1025,2
```

schreibt ein »B« an die zweite Position der ersten Zeile. Im Handbuch sind die Bildschirm-Codes der Zeichen zu finden. Der andere Speicherbereich ab 55296 versieht die Zeichen mit Farbe. In den alten Versionen des C 64 mußten zwei POKEs eingegeben werden, um ein Zeichen sichtbar werden zu lassen. Der zweite POKE diente zum Einschreiben eines Farb-Codes in die dazugehörige Bildschirmfarbspeicherstelle. Hier ist das nicht nötig: Der Speicherbereich ab 55296 wird automatisch mit entsprechenden Werten gefüllt: Die unteren 4 Bit enthalten nach dem Einschalten die Zahl 13, was dem Code der Farbe Hellgrün (vermindert um 1) entspricht. Es steht Ihnen aber frei, bunte Buchstaben zu erzeugen. Dazu ist lediglich ein anderer Farb-Code in die zum Bildschirmspeicher gehörige Farbzelle zu schreiben. Bild 1 soll dieses Zusammenspiel beider Speicherbereiche verdeutlichen:

Übrigens, falls Ihnen die aktuelle Cursorfarbe nicht mehr gefallen sollte, dann können Sie sie durch

```
BANK 0:POKE 241,Farbcode
```

ändern. Auch alle danach geschriebenen Zeichen erhalten die so gewählte neue Farbe.

Nun wissen wir zwar, daß zum Darstellen eines Zeichens auf dem Bildschirm ein Zeichen-Code im Bildschirmspeicher und ein Farb-Code im dazugehörigen Farbspeicher nötig sind, so ganz befriedigend ist das als Erklärung aber noch nicht. Woher weiß unser Computer eigentlich, was er abbilden soll, wenn beispielsweise der Zeichen-Code 1 in der Speicherstelle 1024 steht? Für jedes Zeichen, das der Computer darstellen kann, gibt es im sogenannten Character-ROM ein Muster. Der Zeichen-Code dient als Zeiger auf das erste Byte eines solchen Musters. Die 1 deutet auf das erste Zeichenmuster, das zum großen Buchstaben »A« gehört. Es gibt übrigens auch das nullte Zeichen, den »Klammeraffen«.

Jedem Zeichen sind im Character-ROM 8 Byte zugeordnet. Ein Beispiel zeigt Bild 2.

Daraus ist zu ersehen, daß jedes Zeichen als gesetzte oder nicht gesetzte Punkte in einem 8x8-Rasterfeld defi-

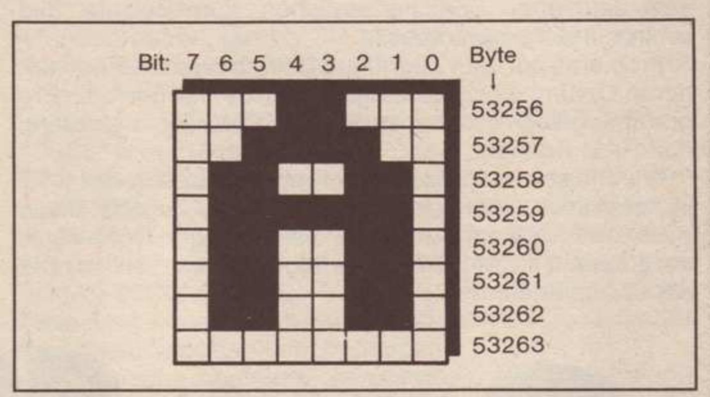


Bild 2. Das Zeichenmuster des Buchstaben A

niert ist. Falls Sie über einen guten Monitor verfügen und Kontrast sowie Helligkeit entsprechend einstellen, dann können Sie diese Punkte auch auf dem Bildschirm erkennen.

Zwei Fragen stellen sich aus dieser Erkenntnis heraus:
- Kann man eigene Zeichen erstellen und verwenden?
- Gibt es eine Möglichkeit, statt jeweils auf solch ein 8x8-Raster auch auf einzelne Punkte einzuwirken?

Die erste Frage wird im Abschnitt über den VIC-Chip beantwortet. Hier sei nur erwähnt, daß es möglich ist.

Die zweite Frage führt uns zu den anderen Grafik-Modi.

Der Hochauflösungsmodus

Wir hatten festgestellt, daß wir im Textmodus (40 Zeichen) immer nur (beispielsweise durch POKE 1024,1) auf ein 8x8-Punkte-Raster zugreifen können. Die Auflösung unseres Bildschirms beträgt in diesem Zustand 40 x 25 Zeichen. Es gibt noch einige Tricks, eine sogenannte Viertelpunktgrafik zu entwerfen, in der die auf der Tastatur vorgegebenen Grafikzeichen (beispielsweise mit <CBM+C> zu erhalten) geschickt verwendet werden. Es juckt aber in den Fingern, jeden Bildpunkt eines solchen 8 x 8-Zeichens einzeln anzusprechen. Das ist tatsächlich mittels dem Grafikmodus 1 (und auch 2) möglich, der durch GRAPHIC1 anwählbar ist. Damit erhöht sich die Auflösung ganz erheblich:

320 horizontale und 200 vertikale Bildschirmpositionen stehen uns hier zur Verfügung. Wir haben den sogenannten Bit-Map-Modus eingeschaltet.

Das Prinzip ist dabei folgendes: Jedem Bildpunkt entspricht ein Bit eines Speicherbereiches, der Bit-Map. Ist in dieser Bit-Map ein Bit gleich 1, dann erscheint an der dazugehörigen Bildschirmposition ein Punkt (das ist wie das Eintragen von Geländeeinzelheiten in eine Landkarte, da-

her der Name Map = Landkarte). Jeweils 8 Bit ergeben ein Byte, woraus sich auf die Größe der Bit-Map schließen läßt: (320*200)/8 = 8000 Byte.

Alles, was auf dem Bildschirm erscheint, ist der Inhalt dieser Bit-Map, die durch den GRAPHIC1-Befehl eingerichtet wird. Wo befindet sie sich? Wenn Sie sich mal vor und nach dem Einrichten der Bit-Map durch PRINT FRE(0) den freien Speicherplatz ausgeben lassen, dann erkennen Sie folgendes: In der BANK 0, also dem Basic-Textspeicher, wird einiges verändert. Normalerweise (also ohne Bit-Map) beginnt der Basic-Text in Speicherstelle $1C00 (dezimal 7168). GRAPHIC1 schiebt den Textstart (und ein eventuell schon im Speicher befindliches Programm) in Windeseile bis $4000 (dezimal 16384). Die Bit-Map beginnt ab $2000 (dezimal 8192).

GRAPHIC1 legt aber nicht nur eine Bit-Map an und schaltet den Videochip auf den speziellen Grafikmodus um, sondern definiert auch ein Koordinatensystem auf dem Bildschirm, das sich aus der Bit-Anordnung des Bildes ergibt. Sehen Sie dazu das Bild 3.

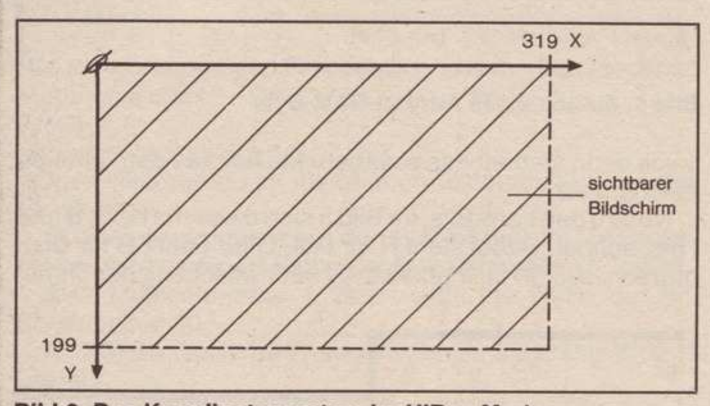


Bild 3. Das Koordinatensystem im HiRes-Modus

Wir finden einen Koordinatenursprung in der linken oberen Bildecke, eine nach unten weisende Y-Achse und eine nach rechts verlaufende X-Achse: Ein solches linkshändiges System bereitet dem Anwender manchmal einige Probleme, weil in der Mathematik ein rechtshändiges System eingesetzt ist (da wäre dann der Ursprung unten links zu finden und die Y-Achse verliefe nach oben). Alle Zeichenbefehle beziehen sich im Normalfall auf die so festgelegten Koordinaten. Ein Befehl

```
DRAW 1,0,0 TO 319,199
```

(zu den Einzelheiten kommen wir noch) zeichnet also eine Linie von links oben nach rechts unten.

Wir haben noch nicht erwähnt, wie der Computer die Farben in diesem Modus festlegt. Das soll beim COLOR-Befehl behandelt werden. Auf Koordinatensysteme kommen wir beim SCALE-Befehl noch einmal zurück.

Die Mehrfarbengrafik

Während in den Grafik-Modi 1 und 2 jedes gesetzte Bit in der Bit-Map einen Bildpunkt nach sich zieht, ist die Mehrfarbengrafik (Modi 3 und 4) immer auf Bit-Paare orientiert. Die Bit-Map hat zwar als Karte des Bildschirminhaltes ihre Bedeutung beibehalten, aber sie wird vom VIC-Chip nun anders gelesen. Es existieren vier Möglichkeiten von Bitkombinationen zu je zwei Bit:

00 Hintergrund
01 Farbe 1
10 Farbe 2
11 Farbe 3

Auf welche Weise welche Farbe angesprochen wird, soll uns beim COLOR-Befehl weiter beschäftigen. Hier interessiert uns die Auswirkung auf die Auflösung.

Weil in der X-Richtung nun nur noch Bit-Paare zählen, halbiert sich die Auflösung. Das Koordinatensystem der

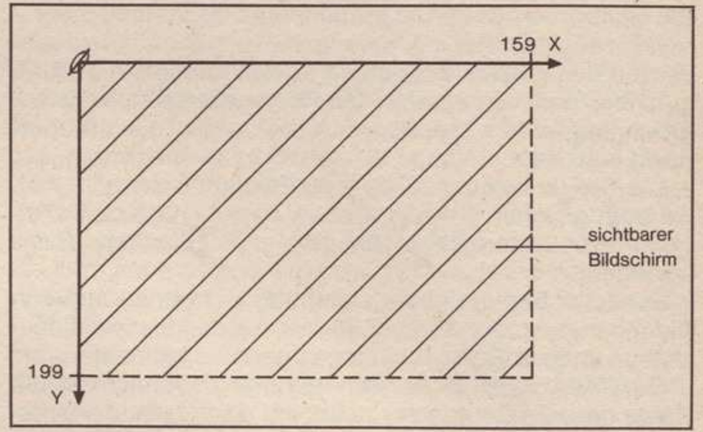


Bild 4. Das Koordinatensystem im Multicolor-Modus

Multicolorgrafik ist ebenso aufgebaut wie das der hochauflösenden Grafik, lediglich in X-Richtung stehen uns nur noch 160 Positionen zur Verfügung. Bild 4 zeigt uns dieses neue Koordinatensystem.

Die Splitscreens

In den Grafik-Modi 2 und 4 wird der Bildschirminhalt durch jeweils zwei verschiedene Betriebsarten des VIC-Chip gewonnen. Im oberen Teil sehen wir den Inhalt der Bit-Map, im unteren aber 40-Zeichen-Text. Dies wird erreicht durch die sogenannte Rasterzeilenunterbrechung. Auch im C64-Modus kann mittels eines Maschinenprogramms diese Aufteilung des Bildschirms programmiert werden. Dabei gab es immer ein Problem, das mit der Bearbeitungszeit zusammenhing. Die Rasterzeilenunterbrechung und die normale Unterbrechung (die 60mal pro Sekunde stattfindet) durften sich nicht gegenseitig in die Quere kommen. Diese Schwierigkeit konnte im Programmbetrieb meistens ganz gut gelöst werden. Im Direktmodus führten aber Cursorbewegungen und die Arbeit mit Sprites oft zu einer zittrigen Grenze der beiden Darstellungsmodi auf dem Bildschirm. Dasselbe Problem scheint im C128-Modus dafür verantwortlich zu sein, daß diese Grenze manchmal geradezu extrem gestört wird, beispielsweise beim Steuern von Sprites über die Splitscreengrenze im Grafikmodus 4.

Beim Einschalten eines Splitscreen befindet sich der Textcursor nicht automatisch im Textbereich. Er muß erst dorthin gefahren werden. Zwei Möglichkeiten bieten sich hier an: Zum einen kann ein PRINT AT simuliert werden durch:

```
BANK 15:SYS 65520,,Zeile,Spalte
```

Zum anderen bietet sich die Definition eines Bildschirmfensters unter Benutzung des WINDOW-Befehls an. Man legt dann einfach den gesamten Textbereich als WINDOW fest. Ist also keine spezielle Trennzeile angegeben worden (dann tritt der Übergang von Grafik zu Text in Zeile 19 ein), kann das durch

```
WINDOW 0,20,39,24
```

geschehen. Übrigens bleibt das Bildschirmfenster auch nach einer Rückkehr in den Textmodus erhalten. Wird das nicht gewünscht, kann durch

```
WINDOW 0,0,39,24
```

der gesamte Bildschirm wieder benutzt werden.

Der 80-Zeichen-Textmodus

Sollten Sie stolzer Besitzer eines 80-Zeichen-Monitors sein, dann können Sie durch GRAPHIC5 die Textausgaben auf dessen Bildschirm lenken. Interessanterweise ist es nun möglich, zwei getrennte Anzeigen zu erhalten, beispielsweise die Multicolorgrafik auf dem 40-Zeichen-Schirm und die Textausgaben auf dem 80-Zeichen-Monitor.

Die Splitscreen-Optionen braucht man dann nicht mehr – sie sind sogar hinderlich, weil sie die Ausgabe von Text wieder auf den 40-Zeichen-Schirm lenken und uns die Mühe machen, den verringerten Grafikausgabebereich zu berücksichtigen (die Y-Achse wird ja nun ab Zeile 19 überdeckt vom Textbereich).

Die Textdarstellung auf dem 80-Zeichen-Schirm folgt etwa dem gleichen Prinzip, wie wir es schon beim 40-Zeichen-Text kennengelernt haben – mit folgenden Unterschieden:

Zunächst benutzt unser Computer jetzt einen anderen Videobaustein, den VDC 8563. Wir werden diesen Videochip an anderer Stelle noch etwas genauer kennenlernen.

Der Bildschirmspeicher ist nirgendwo im Bereich der 128 KByte unseres Computers zu finden, die wir direkt ansprechen können (durch PEEK oder mittels Monitor). Ebensowenig werden wir einen Hinweis auf den Farbspeicher finden. Woran liegt das? Immerhin brauchen wir doch 80 x 25 Byte, also 2000 für das Bildschirm-RAM und ebenso viele Byte für die Farbzellen. Da sind wir auf eine Besonderheit des VDC-Chip gestoßen: Er benutzt einen eigenen 16-KByte-RAM-Bereich, der völlig getrennt vom normalen Speicher existiert und von uns nur über zwei Speicherstellen ansprechbar ist. Welche das sind und wie man sie benutzt, soll ebenfalls beim VDC besprochen werden. Hier nur die Aufteilung dieses Speichers:

0000 bis 2047 Bildschirm-RAM
2048 bis 4095 Attribut-RAM
4096 bis 8191 frei
8192 bis 16385 Zeichenspeicher

Zur Bedeutung des Bildschirm-RAM braucht nichts mehr gesagt zu werden. Das Attribut-RAM entspricht etwa dem Bildschirmfarbspeicher, birgt aber noch mehr Möglichkeiten in sich. Weil wir alle Aspekte der Farbgebung der anderen Modi beim COLOR-Befehl behandeln werden, mit diesem Befehl aber das VDC-Attribut-RAM nicht erschöpfend erfaßt werden kann, soll hier diese VDC-Spezialität erklärt

Besonderheiten des VDC

werden. Zuvor noch ein paar Worte zum Zeichenspeicher. Während im 40-Zeichen-Textmodus ein Zeichenmuster aus dem Character-ROM entnommen wird, ist das im 80-Zeichen-Textmodus nicht der Fall. Vielmehr findet hier schon in der Einschaltphase ein Kopieren aller Zeichen in den speziellen VDC-RAM statt. Von dort stammen nun auch die Zeichenmuster. Sie kennen sicher schon die Erscheinung, daß beispielsweise beim Umschalten auf den DIN-Zeichensatz (mit der CAPS/LOCK-Taste) einige Zeit verstreicht, bis alle Zeichen im neuen Gewand zu sehen sind: Das liegt daran, daß der gesamte Zeichensatz umkopiert werden muß. Im Bereich 8192 bis 16385 finden sich danach nur noch die DIN-Zeichenmuster.

Es wird sicher Ihrer Aufmerksamkeit nicht entgangen sein, daß hier die Rede von RAM ist im Gegensatz zum Zeichen-ROM, das bei der 40-Zeichen-Textdarstellung eine Rolle spielt. Wenn wir wissen, wie man dieses VDC-RAM ansprechen kann, wird uns nichts daran hindern, auch in den Speicherbereich des Zeichensatzes hineinzuschreiben. So können dann eigene Zeichen definiert werden.

Kommen wir nun aber zurück zum Attribut-RAM. Jeder Bildschirmspeicherstelle entspricht ein Byte dieses Attributbereiches. Durch dieses Byte aber wird noch mehr gesteuert als die Farbe, was Sie am Aufbau solch einer Speicherstelle erkennen können (siehe Bild 5). Die Bits 0 bis 3 bestimmen die Farbe, was wir gleich noch untersuchen wollen.

Bit 4: Ist hier eine 1 zu finden, dann blinkt das Zeichen.

Bit 5: Das Zeichen ist unterstrichen, wenn hier eine 1 enthalten ist.

Bit 6: Hiermit wird ein Aspekt der Zeichendarstellung »doppelt gemoppelt«: Die Revers-Darstellung. Eigentlich kann nämlich durch Einschreiben einer »1« an diese Stelle genau dasselbe erreicht werden, wie durch das ansonsten etwas speicherfressende Verfahren eines gesonderten Zeichenmusters, mit dem momentan die inversen Zeichen gezeichnet werden.

Bit 7: Im Gegensatz zum 40-Zeichen-Textmodus, der die Groß- und Kleinschreibung nur durch Umschalten (per CBM + SHIFT-Taste) erlaubt, können im 80-Zeichen-Modus beide Zeichensätze gleichzeitig verwendet werden. Welches von den beiden möglichen Zeichen auf dem Bildschirm zu sehen ist, entscheidet dieses Bit. Liegt hier eine 1 vor, dann stammt das sichtbare Muster aus dem zweiten Zeichensatz.

Bit:	7	6	5	4	3	2	1	0
Name:	ALT	RVS	UL	FL	R	G	B	I

Bild 5. Aufbau eines Attribut-RAM-Byte

Nun zu den Farb-Bits. Im Bild 5 sind diese mit R, G, B und I bezeichnet. Dabei steht R für Rot, G für Grün, B für Blau und schließlich I für Intensität. Je nach Mischung der Signale, die durch unterschiedliche Bitbelegungen der vier Bit erzeugt werden, ergeben sich die 15 möglichen Farbdarstellungen. Bild 6 zeigt alle Kombinationen.

Bit:	3	2	1	0			
	R	G	B	I	dez.	Farbe	entspricht Farbcode im 40-Z.-Modus
	0	0	0	0	0	schwarz	1
	0	0	0	1	1	d'grau	12
	0	0	1	0	2	blau	7
	0	0	1	1	3	h'blau	15
	0	1	0	0	4	grün	6
	0	1	0	1	5	h'grün	14
	0	1	1	0	6	türkis	4
	0	1	1	1	7	orange	9
	1	0	0	0	8	rot	3
	1	0	0	1	9	h'rot	11
	1	0	1	0	10	lila	13
	1	0	1	1	11	violett	5
	1	1	0	0	12	braun	10
	1	1	0	1	13	gelb	8
	1	1	1	0	14	h'grau	16
	1	1	1	1	15	weiß	2

Bild 6. Zuordnung der Bit-Kombinationen zu den Farben (stark vom verwendeten Monitor abhängig!)

All diese einzelnen Aspekte, die das Attribut-RAM erlaubt, können durch CHR$- oder/und ESC-Befehle angesteuert werden. So erzeugt

```
PRINT CHR$(2)"A"
```

ein unterstrichenes und

```
PRINT CHR$(27)+"F";"A"
```

ein blinkendes A. Das Handbuch gibt ab Seite 4-4 erschöpfend darüber Auskunft.

Eine kleine Besonderheit ist allerdings vergessen worden:

Falls Sie einmal <CTRL+S> drücken, während ein Programm läuft, hält dieses sofort an. Damit haben wir eine Pausentaste im Computer. Erst nach einem beliebigen anderen Tastendruck arbeitet der Computer weiter.

GRAPHIC CLR

Eine Version des GRAPHIC-Befehls wurde Ihnen bisher noch verschwiegen. Mit GRAPHIC CLR führen Sie den Computer wieder in den reinen Textmodus zurück. Sie werden sagen, daß dazu doch die Befehle GRAPHIC0 oder GRAPHIC5 ausreichen. Das stimmt aber nur in Grenzen. Falls einmal einer der Grafik-Modi 1 bis 4 eingeschaltet wurde, damit also der ganze Speicher in BANK 0 verändert ist, bleibt die Bit-Map und der Basic-Start bei $4000 erhalten. Das erkennen Sie beispielsweise daran, daß – falls Sie nicht mit einer der Löschfunktionen das Bitmuster gelöscht haben – durch beispielsweise GRAPHIC1 auf dem Bildschirm immer noch das Grafikbild auftaucht, das Sie vorher einmal erstellt haben.

Erst durch den Befehl GRAPHIC CLR krempeln Sie die BANK 0 unseres Speichers wieder völlig um. Der Basic-Start wandert wieder nach $1C00, die Bit-Map wird überschrieben.

COLOR

Wir kommen nun zur Farbgebung in den verschiedenen Modi. Im Befehl

```
COLOR A,B
```

bedeutet A das folgende:

A=0 Hintergrund im 40-Zeichen-Textmodus und in den Grafik-Modi.

A=1 Vordergrundfarbe in den Grafik-Modi. Im Multicolormodus wird damit die Farbe der Bitkombination 01 belegt.

A=2 Multicolorfarbe 1. Das betrifft die Farbe der Bit-Kombination 10.

A=3 Multicolorfarbe 2, die sich auf die Bitkombination 11 bezieht.

A=4 Farbe des Bildschirmrahmens.

A=5 Zeichenfarbe im Textmodus. Damit wird sowohl im 40- als auch im 80-Zeichen-Text die Zeichenfarbe bestimmt.

A=6 Hintergrund des 80-Zeichen-Bildschirms.

B ist der Farbcode. Je nach Fabrikat, eingestelltem Kontrast, Helligkeit und anderen Gerätespezifika variiert die

Farbe	Code	Farbe	Code
Schwarz	1	Hellbraun	9
Weiß	2	Braun	10
Rot	3	Rosa	11
Türkis	4	Dunkelgrau	12
Violett	5	Grau	13
Grün	6	Hellgrün	14
Blau	7	Hellblau	15
Gelb	8	Hellgrau	16

Bild 7. Farben und Code-Nummern

dargestellte Farbe. Der Code und die angegebenen Farben dienen daher eher als ungefährer Richtwert. Bild 7 zeigt die Zuordnung der verschiedenen Werte.

Sehen wir uns nun die Farbgebung etwas genauer an:

Die Farbspeicher

Sowohl einzelne Speicherstellen als auch größere RAM-Bereiche spielen eine Rolle bei der Farbzuordnung. Einen RAM-Bereich haben wir schon beim Textmodus erwähnt: Den Bildschirmfarbenspeicher zwischen 55296 und 56295 ($D800 bis $DBE7). Anders als beim VDC-Attribut-RAM spielen hier nur die Bits 0 bis 3 eine Rolle. Die oberen 4 Bit sind unbenutzt und unterliegen einem ständigen Wandel. Versuchen Sie einmal mittels Monitor und dem Kommando »M FD800« mehrmals hintereinander in dieses Farb-RAM zu sehen. Im Einschaltzustand finden Sie allerlei Werte zwischen 0D und FD, denen aber allen das niederwertige Nibble D eigen ist.

Als weiterer RAM-Bereich liegt – bisher noch nicht erwähnt – in den Grafik-Modi 1 bis 4 ein Farbspeicher für die Bit-Map vor, der von 7168 bis 8167 ($1C00 bis $1FE7) reicht. Parallel zur Einrichtung der Bit-Map sorgen die GRAPHIC-Befehle auch für die Zuordnung dieses Farb-RAM. Wie Sie sehen, enthält der Farbspeicher lediglich 1000 Byte Speicherplatz. Die Farbzuordnung geschieht also weiterhin wie im Textmodus, nämlich Pixel für Pixel (wobei jedes Pixel in der Form des 8 x 8-Rasters gehandhabt wird). So können Sie zwar auch im Hochauflösungsmodus in verschiedenen Farben auf den Bildschirm zeichnen (einfach, indem COLOR1 vor dem Zeichenbefehl eine andere Farbe erhält), sobald aber die Zeichnung genügend dicht an eine andere heranreicht, wechselt an der Stelle das gesamte Pixel seine Farbe, also auch der darin verlaufende Teil der alten Zeichnung.

Der Aufbau der einzelnen Bytes dieses RAM-Bereichs hängt von der Art des eingeschalteten Grafikmodus ab. In der Hochauflösungsgrafik enthält das LSN (das sind die Bits 0 bis 3, also das untere Nibble) einen Code für die Hintergrundfarbe, wohingegen das MSN (die Bits 4 bis 7, das obere Nibble) den Code für die Vordergrundfarbe darstellt. Der verwendete Code ergibt sich jeweils aus dem Farbcode (vorhin als B bezeichnet) minus 1.

Im Mehrfarbenmodus dagegen finden wir im LSN die Multicolorfarbe 1 (als Farbcode-1). Das MSN bleibt unverändert.

Übrigens ist dieser Inhalt des Farb-RAM erst dann vorhanden, wenn auf irgendeine Weise ein Löschvorgang stattgefunden hat, beispielsweise durch GRAPHIC1,1. Vorher – also dann, wenn dieser Speicher lediglich durch GRAPHIC1 eingerichtet, aber nicht gelöscht wurde – können Sie dort nur wirres Byte-Durcheinander erkennen, falls Sie sich mittels dem Monitorkommando M den Speicherbereich ansehen.

Zwei Speicherstellen in der PAGE 3 korrespondieren mit dem eben vorgestellten Farb-RAM:

$3E2 (dezimal994) FG-BG

$3E3 (dezimal995) FG-MC1

FG-BG hängt mit der hochauflösenden, FG-MC1 mit der Multicolorgrafik zusammen. Im Einschaltzustand finden wir in diesen beiden Speicherzellen:

```
$3E2: $DB = bin.1101 1011
                 ↑    ↑
                 13 / 11
            h'grün-1 /d'grau-1
            Vordergr./Hintergr.
$3E3: $D1 = bin.1101 0001
                 ↑    ↑
                 13 /  1
            h'grün-1 / weiß-1
            Vordergr./Multicolor1
```

Mit diesen Werten wird das Farb-RAM belegt, sobald ein Grafikmodus 1 bis 4 eingeschaltet wird und ein Löschvorgang stattfindet. Sobald aber durch einen COLOR-Befehl andere Farben gewählt werden, drückt sich diese Änderung in diesen beiden Speicherstellen aus, und der nächste Löschvorgang belegt das Farb-RAM mit dem Inhalt der entsprechenden Speicherstelle.

Da ergibt sich allerdings ein Problem: Nehmen wir an, wir hätten den Grafikmodus 1 eingeschaltet und wollen nun in den Multicolormodus (3) umschalten. Das ist zwar ohne weiteres möglich, aber die Farben werden erst dann umgeschaltet (also das Farb-RAM erst dann mit dem Code aus $3E3 belegt), wenn ein Löschvorgang stattgefunden hat. Dann ist aber auch die Bit-Map gelöscht! Diese Schwierigkeit kann auf einigen Umwegen gelöst werden. Wie, das werden wir nachher noch sehen.

Vier Zeropage-Speicherstellen spielen in der Farbge-

bung eine wichtige Rolle: $83 bis $86 (dezimal 131 bis 134):
$83 COLSEL aktuelle Farbe
$84 MULTICOLOR1 Multicolorfarbe 1
$85 MULTICOLOR2 Multicolorfarbe 2
$86 FOREGROUND Vordergrundfarbe

Im Einschaltzustand findet man in $83 den Wert 0, in $84 eine 1 (also Farbcode von weiß-1), in $85 steht 2 (das ist rot-1) und in $86 liegt D (13, was hellgrün-1 bedeutet).

COLSEL enthält nach jedem Grafikbefehl die darin angegebene Farbquellenziffer (die wir noch kennenlernen).

Aus dem C64-Modus sind Ihnen zwei Speicherstellen des VIC-Chip sicher bekannt:
$D020 dezimal 53280 Rahmenfarbe
$D021 dezimal 53281 Hintergrundfarbe

Auch im C128-Modus haben die Speicherstellen diese Bedeutung. Im Einschaltzustand finden wir die Farben im LSN:
$D020 enthält $FD = binär 1111 1101
hellgrün-1
$D021 enthält $FB = binär 1111 1011
dunkelgrau-1

Nun kennen wir die beteiligten Speicher. Welche Auswirkungen hat der COLOR-Befehl?

Fassen wir die bis jetzt gewonnenen Erkenntnisse zur Farbgebung der Grafikbildschirme zusammen:

COLOR 0,X belegt den Hintergrund mit der Farbe X. Die Bits 0 bis 3 von $3E2 enthalten den Wert X-1.

Die Bits 0 bis 3 von $D021 werden ebenfalls mit X-1 beschrieben.

COLOR 1,X bestimmt die grafische Vordergrundfarbe. In $86 erscheint X-1.

Die Bits 4 bis 7 der Speicherstellen $3E2 und $3E3 enthalten danach X-1.

COLOR 2,X legt die Multicolorfarbe 1 fest. $84 hat nun X-1 zum Inhalt.

In den Bits 0 bis 3 der Speicherstelle $3E3 befindet sich ebenfalls X-1.

COLOR3,X bestimmt die Multicolorfarbe 2. In $85 befindet sich danach der Wert X-1.

Schließlich wird bei eingeschaltetem Grafikmodus 1 bis 4 durch einen Löschvorgang der Speicherbereich $1C00 bis $1FE7 (dezimal 7168 bis 8167) beschrieben mit
- dem Inhalt von $3E2 bei den Modi 1,2
- oder dem Inhalt von $3E3 in den Multicolor-Modi 3 und 4.

Wir werden nun ausprobieren, welche Bitpaarkombination welcher Farbe im Multicolormodus entspricht. Dazu schreiben wir an den Anfang der Bit-Map drei Zahlen, die diese drei möglichen Kombinationen repräsentieren:

```
binär 1111 1111 = dezimal 255 (Kombination 11)
binär 1010 1010 = dezimal 170 (Kombination 10)
binär 0101 0101 = dezimal  85 (Kombination 01)
```

Die Kombination 00 entspricht der Hintergrundfarbe. Das Programm MULTICOLORTEST1 (Listing 1) fragt zunächst nach den von Ihnen gewünschten Farben.

Die Reihenfolge der Angaben folgt dabei den verschiedenen A-Kennungen des COLOR-Befehls, die wir vorhin verwendet haben. In Zeile 20 werden die von Ihnen gewünschten Farben dann in die Register geschrieben, in Zeile 30 der Mehrfarbenmodus eingeschaltet und das Farb-RAM beschrieben. Die Zeilen 40 bis 60 tragen die oben ermittelten Bitkombinationen in die ersten Plätze der Bit-Map ein. Die folgenden CHAR-Befehle (die lernen wir noch kennen) schreiben nun jeweils in einer der drei Farben einen Orientierungstext auf den Bildschirm. Übrigens sieht der Text manchmal in diesem Modus etwas merkwürdig aus: Die Zeichenmuster sind nämlich nicht für die Bitpaardarstellung entworfen. Aus dem Programm folgt eine recht einprägsame Regel: Der Dezimalwert einer Bitkombination ist die Kennung A des dazugehörigen COLOR-Befehls. Erklärung:

Die Bitkombination 00 wird mittels COLOR0,... farblich festgelegt, die Bitkombination 10 (das ist dezimal 2) folgt in der Zeichnung der Farbe, die durch COLOR2,... definiert wird, und so weiter.

Nun können wir zusammenfassen:

Im hochauflösenden Modus folgt die Farbgebung dem Schema in Bild 8.

Im Mehrfarbenmodus kommt die Hintergrundfarbe (bei

```
5 REM ***** MULTICOLORTEST 1 *****                  <6KQ>
10 INPUT "FARBEN F0,F1,F2,F3";F0,F1,F2,F3           <6D5>
20 COLOR 0,F0: COLOR 1,F1: COLOR 2,F2: COLOR 3,F3   <K47>
30 GRAPHIC 3,1: SCNCLR 3                             <T90>
40 BANK 0: POKE DEC("2000"),255: POKE DEC("2001")
   ,255                                              <0NC>
50 POKE DEC("2002"),170: POKE DEC("2003"),170        <RA4>
60 POKE DEC("2004"),85: POKE DEC("2005"),85          <FJ3>
70 CHAR 1,10,10,"OBEN :11"                           <K2K>
80 CHAR 2,10,13,"MITTE:10"                           <TNU>
90 CHAR 3,10,16,"UNTEN:01"                           <91K>
```

Listing 1. »MULTICOLORTEST 1« – welche Bit-Kombination gehört zu welcher Farbquelle? Bitte mit dem Checksummer 128 (Seite 158) eigeben.

Bit:	7	6	5	4	3	2	1	0
bestimmt:	Vordergrund				Hintergrund			
Bit-Map-Inh.:	1				0			
Farbbefehl:	Color 1,...				Color 0,...			

Bild 8. Ein Byte aus dem Farb-RAM ab $1C00 in den Grafik-Modi 1 und 2

Bit:	7	6	5	4	3	2	1	0
bestimmt:	Vordergrund				Multicolorfarbe 1			
Bit-Kombination:	01				10			
Farbbefehl:	Color 1,...				Color 2,...			

Bild 9. Ein Byte aus dem Farb-RAM ab $1C00 in den Grafik-Modi 3 und 4

00) aus der Speicherstelle 53281, die Farben der Kombinationen 01 und 10 werden gemäß dem Schema in Bild 9 verteilt.

Woher kommt die Farbe der Bitkombination 11? Wir haben festgestellt, daß diese Farbe in der Speicherstelle $85 festgehalten wird. Im C64-Modus wird sie aus dem normalen Text-Farb-RAM (ab $D800) entnommen. Überprüft man aber im C128-Modus diesen Speicherbereich, dann ist darin nur die normale Textfarbe zu finden. Hier muß also ein anderes Prinzip der Farbgebung verfolgt werden. Versuche zeigen, daß auch die Multicolorfarbe 2 pixelweise vergeben wird, was auf ein weiteres Farb-RAM schließen lassen könnte – aber die Suche danach verläuft ergebnislos. Diese Frage kann wohl erst dann geklärt werden, wenn eines Tages ein ROM-Listing des Basic-Interpreters zur Verfügung steht. Solange birgt unser Computer noch einige Geheimnisse.

Farbwechsel

Zwei Aspekte sind es nun noch in bezug auf Farbgebung, die uns interessieren:
- Kann man mehr als vier Farben verwenden?
- Kann man in einem fertigen Bild die Farben ändern?

Der erste Aspekt ist sicher erfüllbar, denn ebenso, wie

das Farb-RAM des Textbildschirms durch POKE-Befehle Zelle für Zelle ansprechbar ist, ist es auch das Grafik-Farb-RAM ab $1C00. Allerdings muß man sich gut überlegen, was man hier hineinschreibt: Sowohl die Bits 0 bis 3 als auch die Bits 4 bis 7 haben ja eine Bedeutung, wie wir gesehen haben.

```
5 REM ***** MULTICOLORTEST 2 *****                     <6KQ>
10 INPUT "FARBEN F0,F1,F2,F3";F0,F1,F2,F3              <6D5>
20 COLOR 0,F0: COLOR 1,F1: COLOR 2,F2: COLOR 3,F3      <K47>
30 GRAPHIC 3,1: SCNCLR 3                               <T90>
40 BANK 0: POKE DEC("2000"),255: POKE DEC("2001")
   ,255                                                <0NC>
50 POKE DEC("2002"),170: POKE DEC("2003"),170          <RA4>
60 POKE DEC("2004"),85: POKE DEC("2005"),85            <FJ3>
70 CHAR 1,10,10,"OBEN :11"                             <K2K>
80 CHAR 2,10,13,"MITTE:10"                             <TNU>
90 CHAR 3,10,16,"UNTEN:01"                             <91K>
100 PRINT "NEUE FARBEN FUER 1 UND 2:"                  <V3S>
110 INPUT "F1,F2=";F1,F2: F1=F1-1: F2=F2-1: F=16*
    F1+F2                                              <5CG>
120 PRINT CHR$(147) CHR$(17)                           <8B7>
130 PRINT "MONITOR" CHR$(17) CHR$(17) CHR$(17) CH
    R$(17)                                             <8CH>
140 PRINT "F 01C00 01FE7 " HEX$(F)                     <4EL>
150 PRINT "X" CHR$(17)                                 <5RN>
160 PRINT "RUN200"                                     <M73>
170 PRINT CHR$(19);                                    <KEN>
180 BANK 0: POKE 842,13: POKE 843,13: POKE 844,13
    : POKE 845,13: POKE 208,4: END                     <TAQ>
190 :                                                  <6SR>
200 PRINT CHR$(147) CHR$(17)"DAS WARS!"                <KEP>
```

Listing 2. »MULTICOLORTEST 2« – ein fertiges Bild ändert seine Farben. Bitte mit dem Checksummer 128 (Seite 158) eigeben.

```
10 REM ***** HIRES-FARBAENDERUNGEN *****               <A99>
20 INPUT "FARBEN F0,F1";F0,F1                          <5M6>
30 COLOR 0,F0: COLOR 1,F1                              <KEF>
40 GRAPHIC 1,1: SCNCLR 1                               <R1T>
50 BOX 1,10,10,100,100,0,1: BOX 0,20,20,30,30,0,1      <UJ0>
60 PRINT "NEUE FARBEN FUER 0 UND 1:"                   <7KU>
70 INPUT "F0,F1=";F0,F1: F0=F0-1: F1=F1-1: F=16*F
   1+F0                                                <B5L>
80 PRINT CHR$(147) CHR$(17)                            <ER7>
90 PRINT "MONITOR" CHR$(17) CHR$(17) CHR$(17) CHR
   $(17)                                               <ESB>
100 PRINT "F 01C00 01FE7 " HEX$(F)                     <KEL>
110 PRINT "X" CHR$(17)                                 <4RJ>
120 PRINT "RUN160"                                     <M33>
130 PRINT CHR$(19);                                    <K6M>
140 BANK 0: POKE 842,13: POKE 843,13: POKE 844,13
    : POKE 845,13: POKE 208,4: END                     <SQS>
150 :                                                  <0S3>
160 PRINT CHR$(147) CHR$(17)"DAS WARS!"                <KEP>
```

Listing 3. »HIRES-FARBAENDERUNGEN« – ein Bild in den Grafik-Modi 1 und 2 ändert seine Farben. Bitte mit dem Checksummer 128 (Seite 158) eingeben.

Eine andere Möglichkeit ist es, mittels des COLOR-Befehls eine oder mehrere Farben (Vordergrund, Multicolor1 oder 2) zu ändern und dann mittels eines Grafik-Befehls eben diese Farbquelle aufzurufen (dazu kommen wir noch). Hier ist zu beachten, daß sich die grafischen Objekte, die damit gezeichnet werden, nicht zu nahe kommen, weil ja immer ein ganzes Pixel umgefärbt wird und daher auch schon vorhandene Bildteile unter Umständen die neue Farbe annehmen.

Wesentlich schwieriger ist es allerdings, ein Bild, dessen Farbgebung uns nicht gefällt, umzufärben. Nehmen wir an, ein Multicolor-Bildschirm trüge Farben, die wir verändern wollen. Was wäre zu tun?

Die Farbe aller Bitpaare 00 läßt sich einfach durch COLOR 0,... umschalten. Andere COLOR-Befehle für die drei restlichen Farben oder auch für die Vordergrund- und die Hintergrundfarbe im hochauflösenden Modus führen zu keinem Ergebnis: Erst ein Löschvorgang ergäbe die Neubelegung des Farb-RAM, wodurch aber auch die Bit-Map frei und unser Bild zerstört würde. Da scheint nur eine Schleife Abhilfe zu bringen, die den Farbspeicher neu schreibt. Nehmen wir an, unsere neue Vordergrundfarbe hieße F1 und die neue Hintergrund- (in den Grafik-Modi 1 und 2) oder Multicolorfarbe 1 (im Grafikmodus 3 oder 4) wäre F2. Dann haben wir zunächst die Gestalt eines Bytes aus dem Farb-RAM zu bestimmen:

```
F1=F1-1:F2=F2-1
```

Das Byte wäre dann:

```
16*F1+F2
```

Die Schleife lautet nun:

```
FOR I=7168 TO 8167:POKE I,16*F1+F2:NEXT I
```

So etwas dauert natürlich eine Weile und wir können ganz gemächlich zusehen, wie von oben nach unten die Farbwechsel stattfinden.

Etwas schneller geht das mit einem Trick, der im Abschnitt zu den selbstmodifizierenden Programmen erklärt wird. Die beiden Programme MULTICOLORTEST2 (Listing 2) und HIRES-FARBAEND (Listing 3) zeigen Ihnen das Verfahren.

Beide Programme zeichnen zunächst etwas auf den Bildschirm und fragen dann nach den gewünschten neuen Farben. Falls Sie nur mit einem Bildschirm arbeiten, sollten Sie noch die Umschaltung auf den Textbildschirm (GRAPHIC0 vor der Frage) und hinterher wieder auf den Grafikbildschirm vornehmen. Nach dem Berechnen des Bytewertes schaltet das Programm den Monitor ein und füllt mittels der F-Funktion den Farbspeicher von $1C00 bis $1FE7. Danach verläßt es den Monitor wieder und fährt mit dem Basic-Programm fort (ab Zeile 160 beziehungsweise 200).

Auf diese Weise können alle beiden Farben im Hochauflösungsmodus verändert werden. Drei Farben von Multicolorbildern sind ebenfalls auf diese Art und mit COLOR 0,... zu erneuern. Interessant ist es übrigens, daß trotz des veränderten Farb-RAM ein Zeichenbefehl (beispielsweise DRAW1,...) immer noch mit der alten Zeichenfarbe arbeitet. Die Erklärung liegt vermutlich darin, daß dazu die Speicherstellen ab $83 Verwendung finden. Wenn man auch die darin befindlichen Angaben der neuen Farbgebung anpaßt, geschieht das nicht mehr.

Wie schon vorhin angedeutet, ist die Änderung der Multicolorfarbe 2, von der noch nicht die Herkunft geklärt werden konnte, ein Problem. Bisher existiert nur eine etwas umständliche Lösung: Speichern Sie Ihre Grafik inklusive des Farb-RAM ab. Das kann mittels des BSAVE-Befehls geschehen:

```
BSAVE"Bild",ON B0,P7168 TO P16191
```

Laden Sie nun das Bild wieder ein mittels:

```
COLOR3,X:GRAPHIC3,1:BLOAD"Bild"
```

In X steht dabei die neue gewünschte Multicolorfarbe 2, die nun gleich beim Einladen berücksichtigt wird.

DRAW

Nach all diesen etwas ermüdenden, aber notwendigen Vorbereitungen kommen wir nun zu den Zeichenbefehlen, deren einfachster der DRAW-Befehl ist.

```
DRAW A,X1,Y1 TO X2.Y2
```

zeichnet eine Linie vom Punkt X1,Y1 zum Punkt X2,Y2 (siehe Bild 10).

A ist eine Kennzahl für die Quelle, aus der die Farbe der Zeichnung herrührt. A hat dabei folgende Bedeutung:
A = 0 Hintergrundfarbe
A = 1 Vordergrundfarbe
A = 2 Multicolorfarbe 1
A = 3 Multicolorfarbe 2

Dieser Befehl ist sehr vielseitig verwendbar. Es ist beispielsweise möglich, einen Linienzug über weitaus mehr

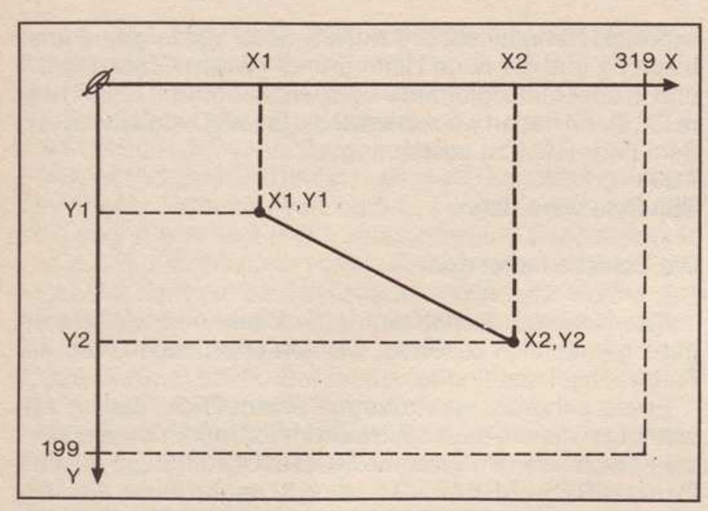


Bild 10. Die Wirkung des DRAW-Befehls

als nur zwei Punkte zu führen. Der Befehl

```
DRAW1,X1,Y1 TO X2,Y2 TO X3,Y3 TO X4,Y4 TO X5,Y5
TO X1,Y1
```

kann so ein grafisches Gebilde wie in Bild 11 erzeugen. Lautet der Befehl

```
DRAW1,X1,Y1
```

dann wird lediglich der Punkt X1,Y1 gezeichnet. Schreibt man dagegen den Befehl als

```
DRAW 1 TO X1,Y1
```

dann dient als Ausgangspunkt die aktuelle Grafikcursorposition. Den Grafikcursor werden wir beim LOCATE-Befehl noch kennenlernen. Von dieser Position aus wird dann eine Linie nach X1,Y1 gezogen.

Nach der Ausführung der verschiedenen DRAW-Befehle befindet sich der Grafikcursor an dem Punkt, der durch die letzte Koordinatenangabe charakterisiert wird.

Läßt man übrigens die Farbkennung weg, schreibt also beispielsweise

```
DRAW ,X1,Y1 TO X2,Y2
```

dann wird die Linie in der Farbe gezeichnet, die im vorangegangenen Zeichenbefehl verwendet wurde. Vielleicht erinnern Sie sich noch an die Speicherstelle $83 COLSEL? Dort befindet sich immer noch vom vergangenen Befehl her die Farbkennung.

BOX

Mit diesem Zeichenbefehl können Rechtecke erzeugt werden. Auch hier sind wieder eine ganze Menge Variationen möglich. Die Syntax des BOX-Befehls ist:

```
BOX A,X1,Y1,X2,Y2,W,F
```

Eine andere Möglichkeit ist es, mittels des COLOR-Befehls eine oder mehrere Farben (Vordergrund, Multicolor1 oder 2) zu ändern und dann mittels eines Grafik-Befehls eben diese Farbquelle aufzurufen (dazu kommen wir noch). Hier ist zu beachten, daß sich die grafischen Objekte, die damit gezeichnet werden, nicht zu nahe kommen, weil ja immer ein ganzes Pixel umgefärbt wird und daher auch schon vorhandene Bildteile unter Umständen die neue Farbe annehmen.

Wesentlich schwieriger ist es allerdings, ein Bild, dessen Farbgebung uns nicht gefällt, umzufärben. Nehmen wir an, ein Multicolor-Bildschirm trüge Farben, die wir verändern wollen. Was wäre zu tun?

Die Farbe aller Bitpaare 00 läßt sich einfach durch COLOR 0,... umschalten. Andere COLOR-Befehle für die drei restlichen Farben oder auch für die Vordergrund- und die Hintergrundfarbe im hochauflösenden Modus führen zu keinem Ergebnis: Erst ein Löschvorgang ergäbe die Neubelegung des Farb-RAM, wodurch aber auch die Bit-Map frei und unser Bild zerstört würde.

Da scheint nur eine Schleife Abhilfe zu bringen, die den Farbspeicher neu schreibt. Nehmen wir an, unsere neue Vordergrundfarbe hieße F1 und die neue Hintergrund- (in den Grafik-Modi 1 und 2) oder Multicolorfarbe 1 (im Grafikmodus 3 oder 4) wäre F2. Dann haben wir zunächst die Gestalt eines Bytes aus dem Farb-RAM zu bestimmen:

```
F1=F1-1:F2=F2-1
```

Das Byte wäre dann:

```
16*F1+F2
```

Die Schleife lautet nun:

```
FOR I=7168 TO 8167:POKE I,16*F1+F2:NEXT I
```

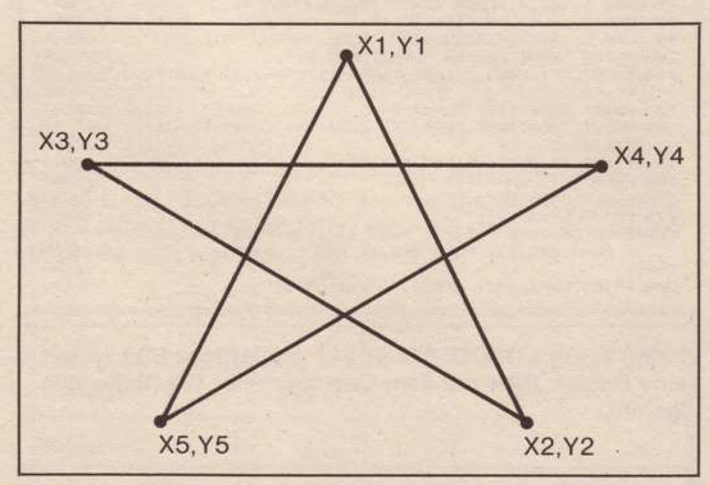


Bild 11. Beispiel einer durch einen DRAW-Befehl erzeugten Figur

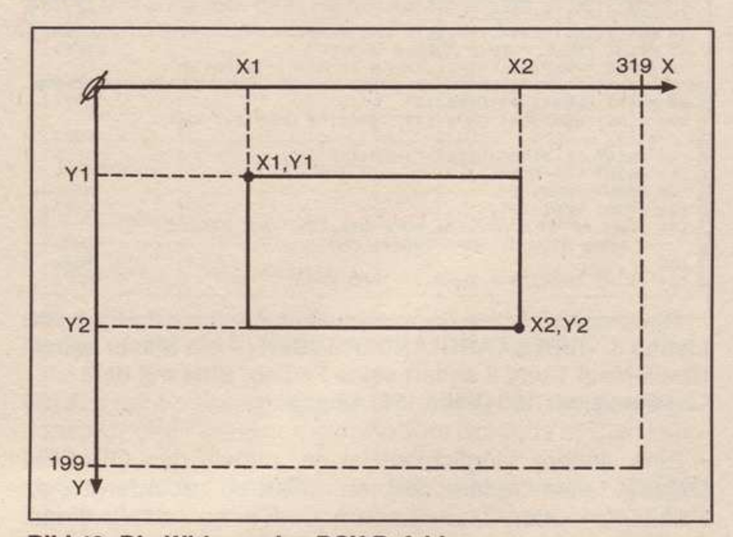


Bild 12. Die Wirkung des BOX-Befehls

So etwas dauert natürlich eine Weile und wir können ganz gemächlich zusehen, wie von oben nach unten die Farbwechsel stattfinden.

Etwas schneller geht das mit einem Trick, der im Abschnitt zu den selbstmodifizierenden Programmen erklärt wird. Die beiden Programme MULTICOLORTEST2 (Listing 2) und HIRES-FARBAEND (Listing 3) zeigen Ihnen das Verfahren.

Beide Programme zeichnen zunächst etwas auf den Bildschirm und fragen dann nach den gewünschten neuen Farben. Falls Sie nur mit einem Bildschirm arbeiten, sollten Sie noch die Umschaltung auf den Textbildschirm (GRAPHIC0 vor der Frage) und hinterher wieder auf den Grafikbildschirm vornehmen. Nach dem Berechnen des Bytewertes schaltet das Programm den Monitor ein und füllt mittels der F-Funktion den Farbspeicher von $1C00 bis

$1FE7. Danach verläßt es den Monitor wieder und fährt mit dem Basic-Programm fort (ab Zeile 160 beziehungsweise 200).

Auf diese Weise können alle beiden Farben im Hochauflösungsmodus verändert werden. Drei Farben von Multicolorbildern sind ebenfalls auf diese Art und mit COLOR 0,... zu erneuern. Interessant ist es übrigens, daß trotz des veränderten Farb-RAM ein Zeichenbefehl (beispielsweise DRAW1,...) immer noch mit der alten Zeichenfarbe arbeitet. Die Erklärung liegt vermutlich darin, daß dazu die Speicherstellen ab $83 Verwendung finden. Wenn man auch die darin befindlichen Angaben der neuen Farbgebung anpaßt, geschieht das nicht mehr.

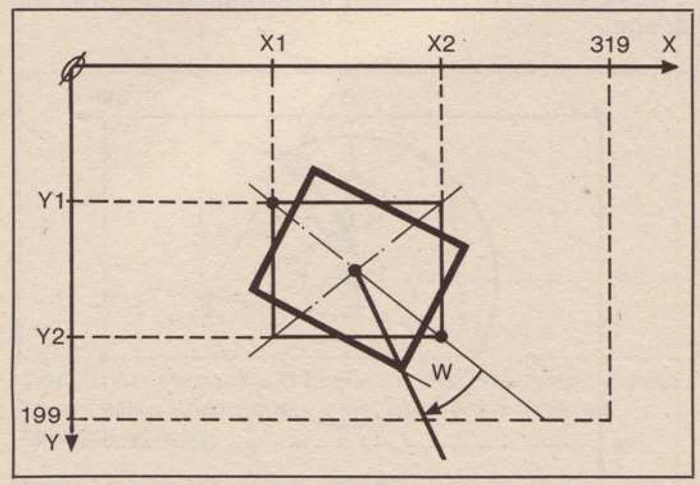


Bild 13. Drehung des Rechtecks durch den Parameter W im BOX-Befehl

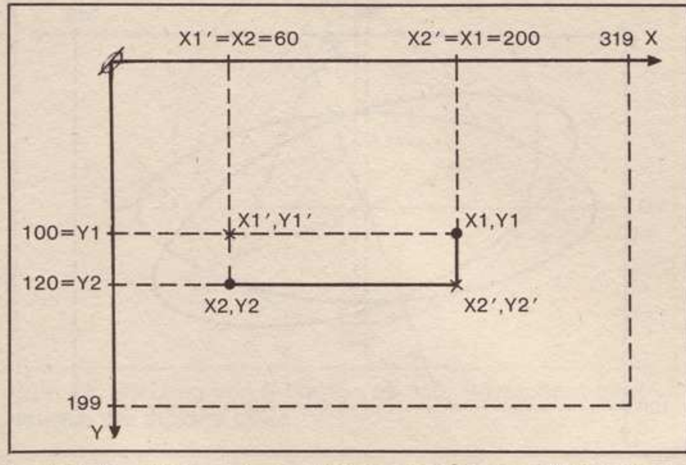


Bild 14. Durcheinandergewürfelte Koordinaten führen dennoch zum richtigen Ergebnis: BOX 1,X2,Y2,X1,Y1 wird gezeichnet wie BOX 1,X1',Y1',X2',Y2'

A ist wieder – wie schon beim DRAW-Befehl – die Farbquelle. Ebenso wie dort kann man A auch weglassen und das Rechteck erhält dieselbe Farbe wie das zuvor gezeichnete Grafikobjekt.

Bild 12 erläutert die Bedeutung der Eckkoordinaten X1,Y1 und X2,Y2.

X1,Y1 sind also die Koordinaten der linken oberen Ecke, X2,Y2 diejenigen der rechten unteren. Wir werden darauf gleich noch einmal zurückkommen.

W ist ein Winkel (in Grad angegeben!), um den das Rechteck zu drehen ist. Die Drehung erfolgt um den Mittelpunkt des Rechtecks im Uhrzeigersinn (siehe Bild 13).

W darf alle Werte zwischen 0 und 65535 annehmen.

F ist eine Kennung, die festlegt, ob das Rechteck ausgefüllt werden soll:

F = 0 Nicht ausfüllen
F = 1 In der gewählten Farbe ausfüllen.

Interessant ist, was geschieht, wenn man Koordinaten eingibt, die unlogisch erscheinen. Beispielsweise in

```
BOX 1,200,100,60,60
```

wurden die Eckpunkte vertauscht. Der Computer wählt sich selbst die richtigen Ecken aus und zeichnet die Gestalt, die wir bei der richtigen Reihenfolge erhalten hätten, also

```
BOX 1,60,60,200,100
```

Würfeln wir die Koordinaten vollends durcheinander (siehe Bild 14).

```
BOX 1,200,100,60,120
```

gibt die linke untere und die rechte obere Ecke an. Gezeichnet wird vom Computer dennoch das richtige Rechteck (mit den Ecken X1',Y1' und X2',Y2'), als hätten wir ihm befohlen

```
BOX 1,60,100,200,120
```

Es scheint, als gäbe es keinen Unterschied bei den verwürfelten Koordinatenangaben. Da existiert aber doch einer! Wir haben bisher den Grafikcursor nicht berücksichtigt. Der steht nämlich immer nach der Zeichnung am Punkt X2,Y2. Das bedeutet, daß er im Fall

```
BOX 1,200,100,60,120
```

auf dem Punkt 60,120 steht, wohingegen er im anderen Fall bei 200,120 zu finden wäre.

Bei einer Drehung des Rechtecks durch einen Wert W wird der Grafikcursor nicht mitgedreht. Er bleibt stur bei X2,Y2 stehen.

Noch nicht ausdrücklich erwähnt wurde die Möglichkeit, für den Fall, daß sie nicht benötigt werden, die Größen W und F einfach wegzulassen. Sie werden dann beide auf 0 gesetzt. Man kann aber auch X2 und Y2 weglassen. Das zu zeichnende Rechteck setzt dafür einfach wieder die Koordinaten des Grafikcursors ein.

Ein Beispiel:

```
BOX 3,100,50,200,80,30,1
```

zeichnet in der Multicolorfarbe 2 (dazu muß natürlich der Grafikmodus 3 eingeschaltet sein) ein Rechteck, das um 30 Grad um seinen Mittelpunkt im Uhrzeigersinn gedreht wurde und ausgefüllt ist.

CIRCLE

Ein geradezu phantastisch vielseitiger Befehl, nicht nur zum Zeichnen von Kreisen, wie es der Name (circle ist das englische Wort für Kreis) suggeriert. Was man damit alles anstellen kann, werden Sie gleich noch sehen. Wieder kann man über eine lange Parameterliste alle Optionen ausschöpfen. Aber keine Angst, es müssen bei weitem nicht immer alle angegeben werden. Komplett sieht der Befehl so aus:

```
CIRCLE A, XM,YM, RX,RY, W1,W2, W, S
```

Der erste Parameter A dient wie schon bei den anderen Zeichenbefehlen zur Festlegung der Farbquelle.

Die Koordinaten XM,YM legen den Mittelpunkt des Objekts fest, das wir zu zeichnen gedenken. Soll es ein Kreis werden, ist es also der Kreismittelpunkt.

RX und RY sind die Halbmesser in horizontaler und vertikaler Richtung. Gedanklich geht man dabei von der Ellipse aus, die ja zum Kreis wird, wenn beide Radien gleich sind. In diesem Fall ist RX = RY und man kann sogar RY weglassen. Bild 15 zeigt, wie die bisher verwendeten Bezeichnungen zu verstehen sind.

Zwei CIRCLE-Befehle erzeugen hier einen Kreis (CIRCLE 1,XM,YM,RX) und eine Ellipse (CIRCLE 1,XM,YM, RX',RY).

Die Angaben W1 und W2 erlauben die Festlegung eines Bogens aus dem Gesamtobjekt, der zu zeichnen ist. Aus-

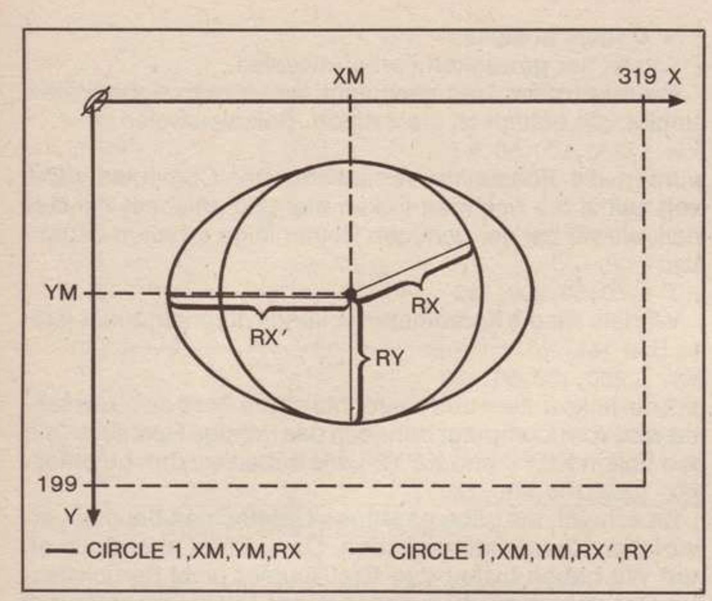


Bild 15. Wirkung der Befehle CIRCLE 1,XM,YM,RX und CIRCLE 1,XM,YM,RX',RY

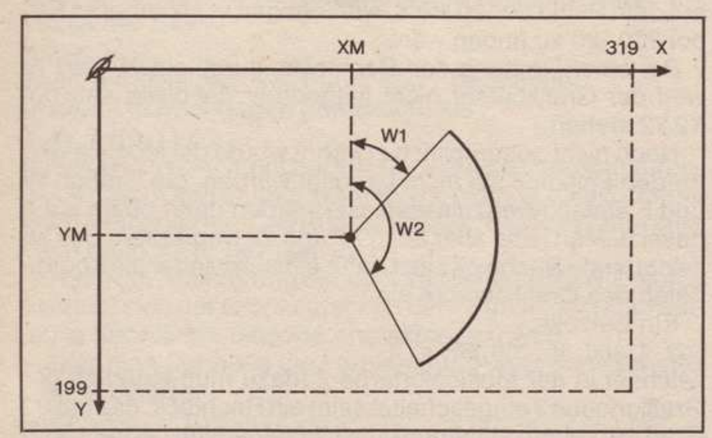


Bild 16. Zeichnen eines Kreis-(oder Ellipsen-)Bogens durch CIRCLE 1,XM,YM,RX,RY,W1,W2

gangspunkt für die Messung der Winkel W1 und W2 ist der obere Scheitelpunkt der Ellipse, die man sich durch die Figur gelegt denken kann. Die Formulierung hört sich wesentlich einfacher an, wenn man von der Zeichnung eines Kreises ausgeht, der im normalen Koordinatensystem liegt. Dann zählt der Winkel vom obersten Kreispunkt an im Uhrzeigersinn, was Bild 16 erläutern soll.

W1 und W2 dürfen ohne Fehlermeldung alle Werte zwischen 0 und 65535 annehmen, wobei allerdings Angaben, die größer als 33024 sind, allerlei Unsinn auf dem Bildschirm erzeugen.

Was geschieht, wenn W1 größer als W2 gewählt wird? Kein Problem: Das Ergebnis finden Sie in einem Beispiel in Bild 17:

Es wird der Bogen von W1 über 0 hinweg bis W2 gezeichnet.

Ein weiterer Parameter ist W, der Drehwinkel. Ebenso wie beim BOX-Befehl kann auch das durch CIRCLE erzeugte Objekt um den Mittelpunkt im Uhrzeigersinn gedreht werden. Bild 18 zeigt die Verhältnisse dabei.

Ohne zu murren, nimmt der Computer Werte zwischen 0 und 65535 für W entgegen.

Der letzte Parameter S kann zwischen 1 und 255 groß sein. Er gibt an, nach welchem Winkel jeweils der nächste Kreis- oder Ellipsenpunkt zu berechnen ist. Die so ermittelten Punkte werden durch Geraden miteinander verbunden. Niedrige Werte von S bedeuten kleine Winkel. So ist der Wert S=2 voreingestellt, und die miteinander verbundenen Eckpunkte ergeben den Eindruck eines Kreises (oder einer Ellipse). Erhöht man aber S, dann beginnt der Kreis (ich erspare mir im weiteren den Hinweis auf die Ellipse) allmählich kantig zu werden. Man kann es zwar nicht erkennen, trotzdem ist der schnell gezeichnete Kreis (oder...) in Wahrheit ein 36-Eck, wenn S=10 gewählt wurde. Weitere Polygone ergeben sich zum Beispiel:

```
S=1  360-Eck
S=2  180-Eck
S=10 36-Eck
S=20 18-Eck
S=30 12-Eck
S=60  6-Eck
```

Das Bild 19 soll dazu dienen, diese Angelegenheit zu illustrieren.

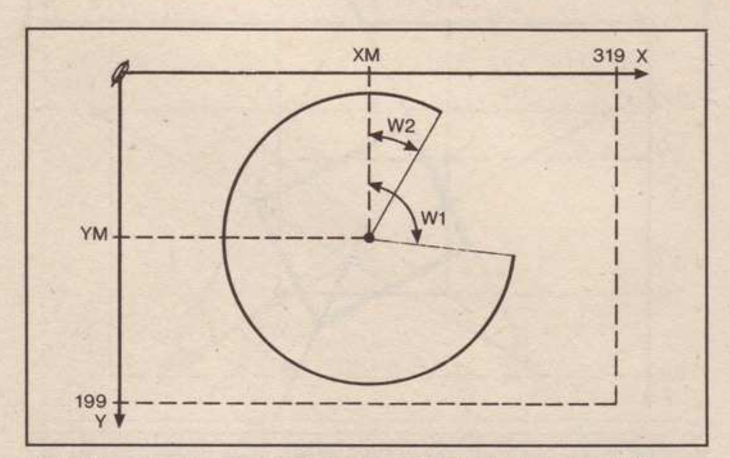


Bild 17. Zeichnen des Bogens, wenn W1 größer als W2

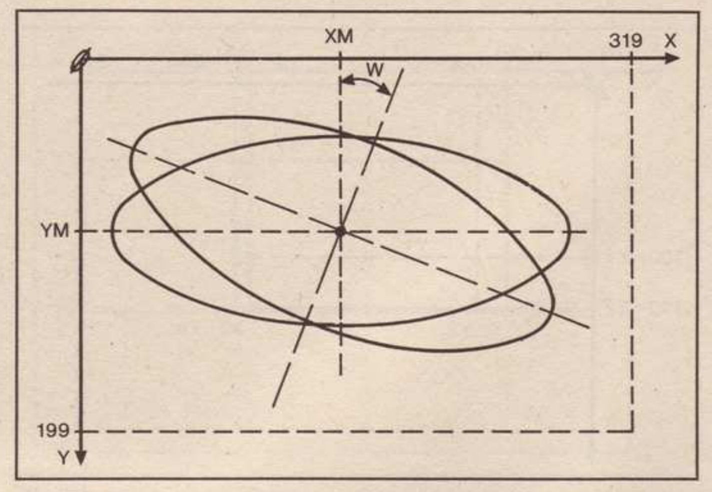


Bild 18. Drehung der Ellipse um den Winkel W um den Mittelpunkt XM,YM durch: CIRCLE 1,XM,YM,RX,RY,,,W

Weil der gesamte Kreis einen Winkel von 360 Grad überstreicht, läßt sich eine Formel angeben, mit der gezielt n-Ecke erzeugt werden können:

S = 360/n

Wenn Sie also ein Dreieck (n=3) erzeugen möchten, müssen Sie S=120 eingeben. Die Formel zeigt uns noch zwei weitere Eigenschaften von S:

- Zum einen kann man sie umdrehen und berechnen, wie viele Ecken ein Polygon bei einem bestimmten S-Wert hat:

  n = 360/S

  Dabei zeigt es sich, daß für »n« auch allerhand gebrochene Zahlen zustandekommen. Weil es aber beispielsweise kein 7,2-Eck gibt (bei S=50), muß es sich hier um ein krummes 7-Eck handeln.
- Zum zweiten sind Eingaben größer als 180 sinnlos.

S=180 führt zum »Zweieck«, also zum Durchmesser des Kreises (siehe Bild 20).

Der Startpunkt ist immer oben (also bei einem Winkel von 0 Grad) zu finden, der Endpunkt liegt auf der Kreisperipherie um 180 Grad demgegenüber verdreht. Ein Weiterzählen um noch einmal 180 Grad führt wieder zum Startpunkt. Wächst S auf noch höhere Werte, dann wird nur noch eine Sehne gezeichnet vom Startpunkt zum Punkt 1, der beispielsweise bei S=225 um 225 Grad, bei S=255 um 255 Grad verdreht auftritt. Ein weiterer Punkt wird hier nicht mehr berechnet, weil damit der Ausgangspunkt bei 0 Grad überschritten würde.

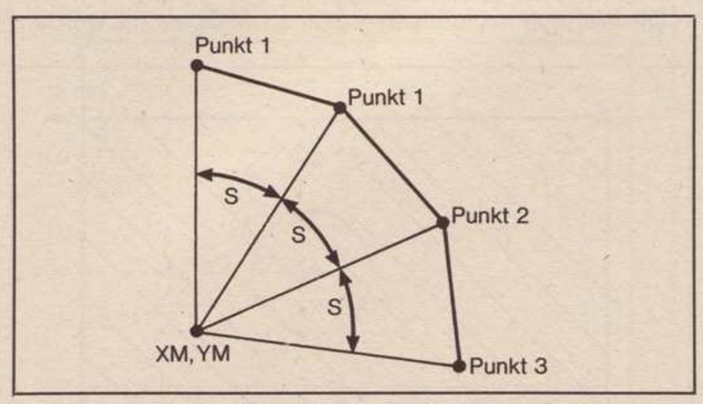


Bild 19. Der Parameter S bestimmt den Winkelabstand der berechneten Kreispunkte, zwischen denen Geraden gezogen werden

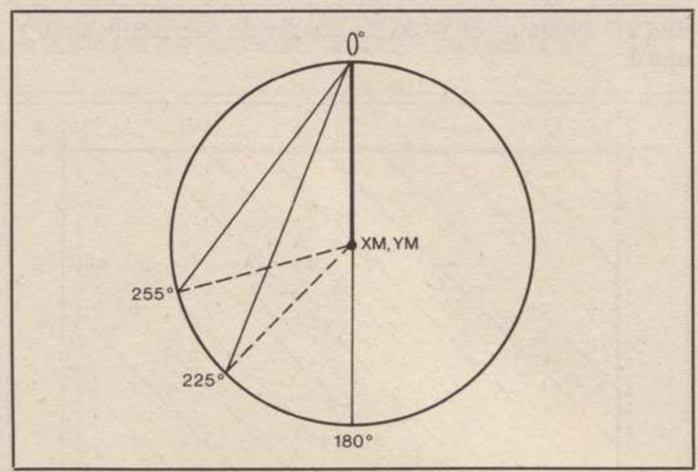


Bild 20. Wirkung von S-Werten ab 180. Gezeichnet wird jeweils die dickere Linie

Den Grafikcursor hätten wir fast noch vergessen: Der befindet sich (im Gegensatz zum BOX-Befehl) immer dort, wo der letzte Punkt des CIRCLE-Befehls gesetzt wurde.

Übrigens gilt es noch eine Besonderheit des CIRCLE-Befehls im Multicolormodus zu beachten. Gibt man hier nämlich einen Befehl zum Zeichnen eines Kreises in der Form

```
CIRCLE 1,XM,YM,RX
```

ein, wobei ja automatisch RY = RX gesetzt wird, dann entsteht zwar auf dem Bildschirm ein Kreis. Der ist aber falsch, was Sie unschwer erkennen können, indem Sie oben und unten Tangenten anlegen. Das Ganze sähe beispielsweise so aus:

```
CIRCLE 1,80,100,40:DRAW 1,0,40 TO 159,40
DRAW 1,0,120 TO 159,120
```

Als gebildete Programmierer sollten wir erwarten, daß diese Geraden als Tangenten den Kreis am oberen und am unteren Rand berühren. Aber weit gefehlt, sie schneiden ihn! Probieren wir dagegen einmal, den Kreis mittels

```
CIRCLE 1,80,100,40,40
```

zu zeichnen, dann sollte ja dasselbe herauskommen. Aber das Ergebnis dieser Zeile ist eine Ellipse! Woran liegt das? Die Ursache dafür liegt in der ungleichen Skalierung der X- und der Y-Achse im Multicolormodus. Wenn wir dem Computer den Auftrag erteilen, einen Kreis zu zeichnen, dann zeichnet er ihn – ohne Rücksicht auf Verluste oder Skalierungen oder ähnliche Feinheiten.

Damit kann eine von Kurvenzügen umschlossene Fläche ausgefüllt werden. Sie muß aber auch wirklich völlig eingerahmt sein, denn das kleinste Loch in der Hülle führt dazu, daß auch die benachbarte Fläche bemalt wird. Im Befehl

```
PAINT A, X1,Y1, M
```

haben die einzelnen Größen folgende Bedeutung:

A ist – wie schon gehabt – wieder die Farbquelle, mit der das Ausmalen geschehen soll.

X1 und Y1 sind die Koordinaten eines Punktes, der in der

PAINT

auszufüllenden Fläche enthalten ist. Gleichzeitig ist das der Startpunkt der Aktion und hinterher liegt genau hier der Grafikcursor.

Ein etwas schwerer verständlicher Parameter ist der Modus M. Ist nämlich M gleich 0, dann wird solange ausgefüllt, bis eine Umrahmung in der Farbe angetroffen wird, die im Parameter A gewählt wurde. Ist dagegen M gleich 1, dann zählt auch jede andere Farbe der Umrandung – sofern sie nur nicht die des Hintergrundes ist – als Grenze des Ausmalens. Befinden wir uns beispielsweise im Grafikmodus 3, dann passiert bei der Befehlskombination:

```
BOX 2,10,10,100,100:CIRCLE 3,80,100,40: PAINT 1,85
,85
```

nicht das Ausmalen der Schnittmenge von Kreis und Rechteck, sondern ungerührt um unsere Bemühungen übermalt der Computer alles, was auf dem Bildschirm zu sehen war. Der Grund ist, daß keine Umrahmung in der Farbe mit der Kennung A = 1 vorhanden war (BOX lief mit A=2 und CIRCLE mit A=3) und wir den Modus 0 gewählt haben. Modus 1 hätte unseren Wunsch erfüllt.

PAINT arbeitet im Modus M=1 nicht, wenn am Startpunkt schon eine andere Farbe als die Hintergrundfarbe gefunden wird. Im Modus M=0 dagegen wird immer dann ausgefüllt, wenn dort eine Farbe vorhanden, die von der ausgewählten verschieden ist.

CHAR

Eigentlich kein Grafikbefehl im engeren Sinn ist dieser Befehl zum Drucken eines Textes. Seine interessanteste Anwendung ist aber die Beschriftung von Grafiken, was auf andere Weise (außer in den Splitscreen-Modi) nicht möglich ist.

```
CHAR A, XT,YT, A$, R
```

bildet einen Text A$ in der Farbe der Quelle A ab. Das erste Zeichen befindet sich bei den Koordinaten XT,YT. Das Koordinatensystem, das hier eine Rolle spielt, ist das im Textmodus gültige (25 Zeilen zu je 40 Spalten oder aber 80 Spalten). A$ kann jede beliebige Kombination von gültigen Zeichen enthalten. In den Grafik-Modi werden allerdings die Steuerzeichen als grafische Zeichen mitgedruckt.

R ist eine Kennzahl, die es erlaubt, den Text normal oder revers darzustellen:

R = 0 Normaldarstellung
R = 1 Reversdarstellung

Sollte der Text nicht in einer Zeile Platz finden, dann wird er in der nächsten fortgesetzt, also ein Verhalten, wie wir es

auch vom PRINT-Befehl her kennen.

Im Multicolormodus sehen die Zeichen häufig etwas merkwürdig aus. Die Zeichenmuster sind ja nicht zur Darstellung im Bitpaarmodus konstruiert. Soll ein Text in der Multicolorfarbe 2 gedruckt werden, dann ist A auf 0 und R auf 1 zu setzen. Soll dagegen die Multicolorfarbe 1 verwendet werden, hält man die Farbquelle A auf 1 und für R wählt man den Wert 0. So steht es jedenfalls im Handbuch (auf den Seiten 4-28 und 4-29). Zumindest mein C 128 folgt diesem Rezept nicht, sondern es genügt, einfach A entsprechend der gewünschten Farbquelle (also 0,1,2,3) zu wählen.

Der Grafikcursor bleibt übrigens vom CHAR-Befehl unbeeindruckt.

LOCATE

Wir haben bei den bisher untersuchten Befehlen immer wieder einen Grafikcursor und dessen Verhalten bei den verschiedenen Operationen erwähnt. Sie haben diesen Cursor aber noch nie zu Gesicht bekommen – werden Sie auch nie, denn er führt sein verborgenes Leben nur in einigen Speicherstellen, nie aber auf dem Bildschirm. Trotzdem ist er recht nützlich. Er dient gewissermaßen als Merkstelle für bestimmte Punktkoordinaten:

- Als Startpunkt bei DRAW
  Beispielsweise für DRAW 1 TO X2,Y2
- Als Eckpunkt bei BOX
- Als Mittelpunkt bei CIRCLE: Laut Handbuch soll man die Koordinaten XM,YM weglassen können und dafür werde automatisch der Ort des Grafikcursors eingesetzt. Leider war das bei meinem C 128 nicht nachvollziehbar. Es ergab sich nur allerlei Unsinn auf dem Bildschirm.
- Als Startpunkt für PAINT: Auch hier kann man ohne Angabe der Startpunktkoordinaten Flächen ausmalen, denn es wird automatisch der Ort des Grafikcursors verwendet.

LOCATE X,Y führt diesen Grafikcursor an den durch die Koordinaten X,Y definierten Punkt.

RDOT

Wollen Sie erfahren, wo sich der Grafikcursor im Augenblick befindet, dann dient dieser Befehl Ihrem Wunsch. Das Argument von RDOT(n) hat folgende Bedeutungen:
n = 0 liefert die aktuelle X-Position.
n = 1 ergibt die aktuelle Y-Position des Grafikcursors.
n = 2 macht eine Aussage über die aktuelle Farbquelle.

Die Koordinatenwerte, die RDOT-Anfragen ergeben, beziehen sich immer auf die grundlegenden Koordinatensysteme. Im hochauflösenden Grafikmodus stammen die Werte daher immer aus einem System, dessen X-Werte zwischen 0 und 319 liegen und dessen Y-Werte von 0 bis 199 reichen. Im Multicolormodus geht die X-Achse nur bis 159. Das im Auge zu behalten ist besonders für den nun folgenden Befehl wichtig:

SCALE

Mittels
```
SCALE n, Xmax, Ymax
```
können wir neue Koordinatensysteme auf unserem Bild-
X von 0 bis 319, Y von 0 bis 199
schirm definieren. Die Kennzahl n kann zwei Optionen anwählen:
n = 0 Das reguläre System ist eingeschaltet:
Grafik-Modi 1 und 2:
n = 1 Ein verändertes System ist nun Bezugssystem geworden.

Wenn durch n=1 die neue Skalierung eingeschaltet wurde, gibt es zwei Möglichkeiten. Schreibt man lediglich SCALE 1, dann hat man ein Koordinatensystem gewählt, dessen X- und Y-Koordinaten jeweils von 0 bis 1023 reichen. Der zweite Weg ist es, selbst die Reichweiten der Skalen festzulegen. Das geschieht durch die Angaben von Xmax und Ymax, den Höchstwerten auf der jeweiligen Achse. So wählt im Grafikmodus 1 der Befehl
```
SCALE 1, 400, 300
```
ein System, dessen X-Achse von 0 bis 400 und Y-Achse von 0 bis 300 reicht (siehe Bild 21).

Die Werte Xmax müssen größer als 320 gewählt werden,

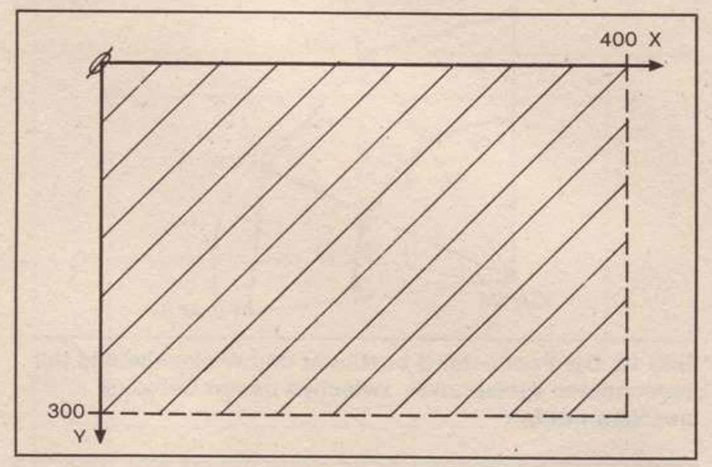


Bild 21. Wirkung von SCALE 1,400,300 in den Grafik-Modi 1 und 2

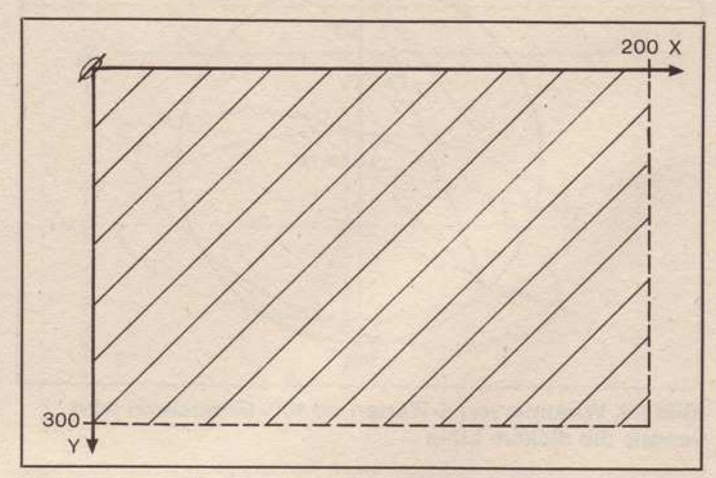


Bild 22. So wirkt SCALE 1,400,300 in den Grafik-Modi 3 und 4

die für Ymax größer als 200. Es ist so also nicht möglich, verkleinerte Systeme zu erzeugen. Im Gegensatz zu den Handbuchangaben, die nur Werte bis 1023 erlauben, ist es tatsächlich möglich, sowohl Xmax als auch Ymax bis 32767 anzugeben.

Im Multicolormodus ist zu beachten, daß weiterhin auf der X-Achse immer Bit-Paare dargestellt werden. Im Grafikmodus 3 oder 4 erzeugt daher der Befehl
```
SCALE 1, 400, 300
```
ein System, dessen X-Koordinaten von 0 bis 200 und dessen Y-Koordinaten von 0 bis 300 reichen (siehe Bild 22).

Natürlich werden weiterhin die Abbildungen auf dem Bildschirm mit einer Auflösung von insgesamt 64000 Punkten (oder 32000 im Multicolormodus) dargestellt. Die Bit-Map bleibt unverändert. Lediglich der Berechnungsweg zur Darstellung der Punkte verändert sich. Etwas unglück-

lich werden viele Anwender des C 128 darüber sein, daß der SCALE-Befehl keine weitergehenden Umstellungen des Koordinatensystems ermöglicht. Wünschenswert wäre sicherlich die Möglichkeit gewesen, den Ursprung zu verschieben (wie es der TRS-Befehl der Grafikerweiterung HIRES-3 für den C 64-Modus erlaubt). Durch diese mindere Ausstattung kommt dem SCALE-Befehl nicht die große Bedeutung zu, die er haben könnte, denn jeder Anwender muß die langwierigen Transformationen weiterhin programmieren.

SCALE hat auch Einfluß auf das Ergebnis eines CIRCLE-

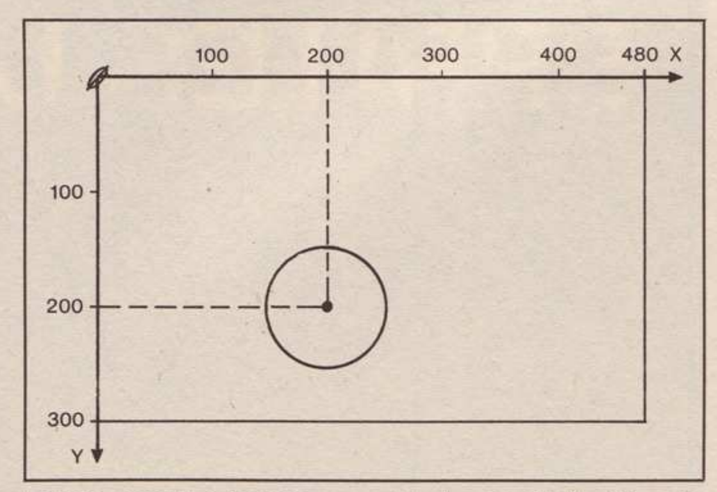


Bild 23. SCALE 1,480,300 erzeugt in den Grafik-Modi 1 und 2 dasselbe Achsenverhältnis wie das normale System: 320/200 = 480/300 = 1.6. Der Kreis CIRCLE 1,200,200,50 ist identisch mit dem Kreis CIRCLE 1,200,200,50,50.

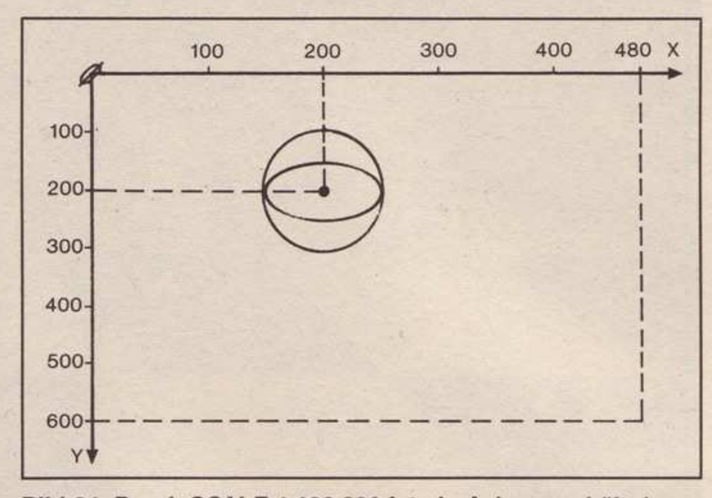


Bild 24. Durch SCALE 1,480,600 ist ein Achsenverhältnis von 480/600 = 0.8 geschaffen worden. CIRCLE 1,200,200,50 erzeugt den falschen Kreis. CIRCLE 1,200,200,50,50 zeichnet zwar den richtigen Kreis, der aber zur Ellipse verzerrt wird.

Befehls. Ebenso, wie wir es vorhin beim Multicolormodus beobachten konnten, wo ohne Rücksicht auf den anderen Maßstab der Y-Achse ein Kreis gezeichnet wurde, findet das hier in allen Grafik-Modi statt. Wenn also der richtige Kreis – das heißt derjenige, der bei allen Maßstäben in Y-Richtung den gleichen Durchmesser aufweist wie in X-Richtung – gezeichnet werden soll, muß das nun immer durch den CIRCLE-Befehl mit zwei Radienangaben geschehen. Das sollen Ihnen die Bilder 23 und 24 erläutern.

Die Verhältnisse in diesen Abbildungen beziehen sich auf die Grafik-Modi 1 und 2. Das Achsenverhältnis im Multicolormodus ist durch den Faktor 2 in X-Richtung verändert. Man sollte also festhalten, daß immer dann unverzerrte Bilder zustandekommen, wenn die Achsenverhältnisse wie folgt sind:

Grafik-Modi 1 und 2:X:Y = 1.6
Grafik-Modi 3 und 4:X:Y = 3.2

Der SCALE-Befehl kann – nebenbei bemerkt – innerhalb eines Programms oder bei ein und demselben Bildschirm – mehrfach angewendet werden. Das ergibt interessante Effekte.

SCNCLR

»SCreeN CLeaR« ist englisch und heißt etwa »Bildschirm freimachen«. Das bedeutet, daß in den Text-Modi der Speicherbereich ab $400, in den Grafik-Modi sowohl die Bit-Map als auch das Farb-RAM ab $1C00 gelöscht, beziehungsweise mit Leerzeichen oder Farbcode beschrieben werden. Welche Bereiche gelöscht werden, welche Bildschirme also, entscheidet das Argument von SCNCLR n. Die Zuordnungen von »n« sind hier die gleichen, die auch beim GRAPHIC-Befehl verwendet werden.

Ein gerade sichtbarer Grafik-Bildschirm ist auch löschbar ohne Angabe eines Arguments, SCNCLR genügt in diesem Fall. Sollte im Textmodus ein Fenster mittels WINDOW definiert sein, dann bezieht sich der Löschbefehl auf den Fensterinhalt.

Bei der Anwendung des SCNCLR-Befehls in den Grafik-Modi muß bedacht werden, daß für den Multicolormodus und für den hochauflösenden Modus dieselbe Bit-Map benutzt wird. Nach SCNCLR3 ist also auch ein im Grafik-Modus 1 sinnvolles Bild verschwunden.

WIDTH

»Width« ergibt aus dem Englischen ins Deutsche übersetzt den Begriff »Breite«. Gemeint ist damit die Strichstärke, in der Zeichenbefehle auf dem Bildschirm realisiert werden.

Mit WIDTH 1 schaltet man die normale Strichdicke ein, mit WIDTH 2 wird allen grafischen Objekten die doppelte Strichbreite zugeordnet.

Alle Zeichenbefehle nach einem WIDTH-Befehl führen dann zur darin angegebenen Strichstärke.

RCLR

Dieser Befehl bildet das Gegenstück zum COLOR-Befehl. Mittels RCLR n ist es möglich, festzustellen, welche Farbzuordnungen zu den verschiedenen Quellen getroffen worden sind. »n« ist dabei eine Kennzahl mit derselben Bedeutung wie der erste Parameter der COLOR-Anweisung.

```
PRINT RCLR(2)
```

gibt also Auskunft über die Farbe, die als Multicolorfarbe 1 definiert wurde.

RGR

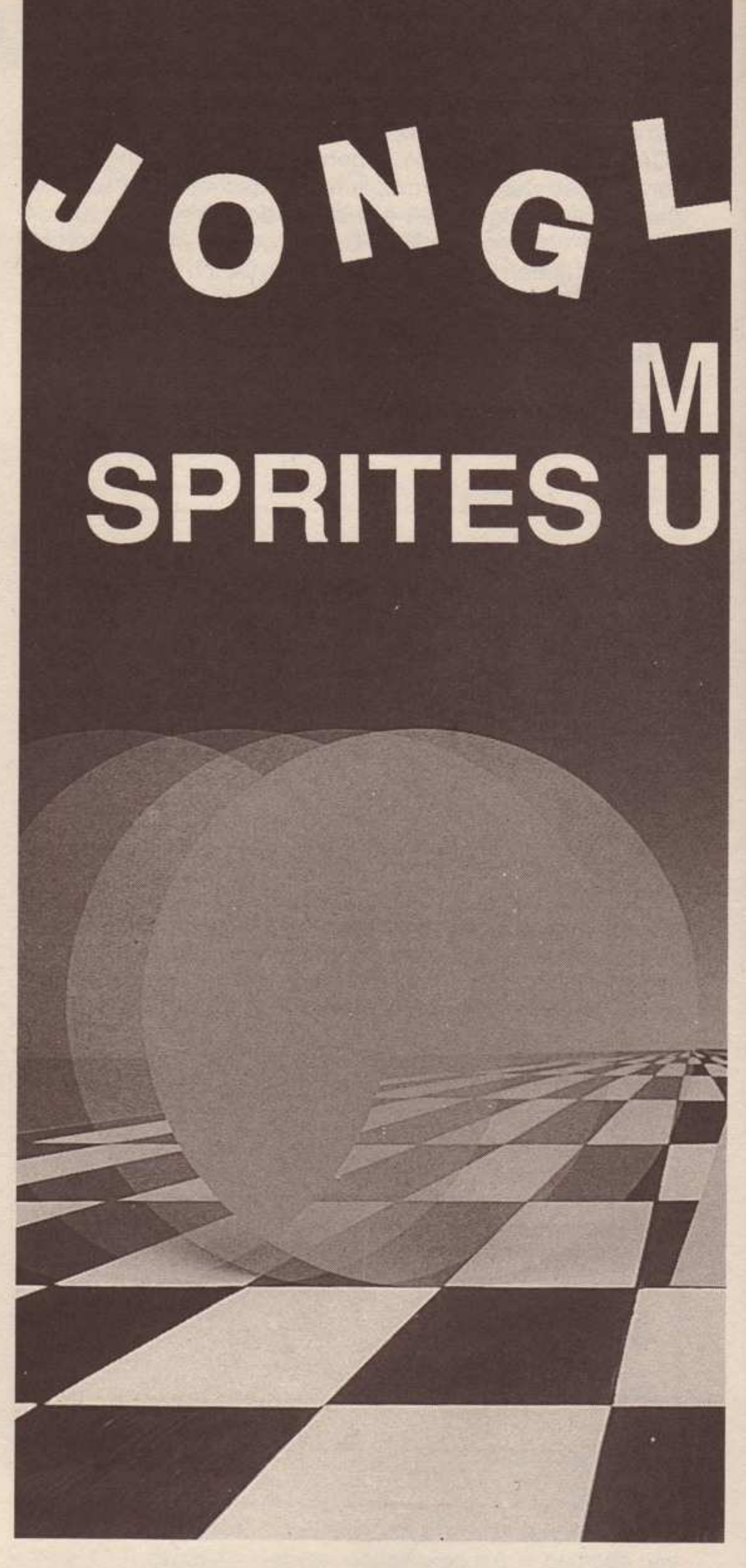
Vermutlich relativ selten benötigt, ergibt

```
PRINT RGR(x)
```

den aktuellen Grafikmodus. Das Argument x ist hier lediglich ein Dummy, hat also keine Funktion außer der eines Platzhalters. Die sich ergebende Zahl entspricht dem ersten Argument des GRAPHIC-Befehls, gibt also den Grafikmodus an.

Mit diesen Grafikbefehlen haben Sie also ein Werkzeug in der Hand, das bei richtiger Anwendung die tollsten Effekte auf dem Bildschirm ermöglicht.

(Heimo Ponnath/kn)



Um die Programmierung der Sprites auf dem C64 zu erklären, ist schon viel Druckerschwärze verbraucht worden. Fast schien es eher die Aufgabe von Assembler-Programmierern zu sein, durch die POKE-Wüsten und Zahlensysteme zu gelangen um schließlich das Ziel – ein bewegtes Sprite auf dem Bildschirm - zu erreichen. Im C128-Modus erspart uns das Basic 7.0 all diese Strapazen: Sprites zu erstellen und zu verwalten wird zum Vergnügen. Nachfolgend lernen wir die dazu nötigen Befehle bis in ihre Feinheiten kennen. Außerdem wird uns eine neue Gruppe von Grafikobjekten beschäftigen: die Shapes.

Sollten Sie Ihnen noch nicht begegnet sein, die kleinen Kobolde (»sprite« ist englisch und heißt auf deutsch »Kobold« oder »Gespenst«), dann seien sie hiermit kurz vorgestellt.

Es handelt sich um bewegliche grafische Objekte (daher auch der Ausdruck MOB=movable objects, der manchmal verwendet wird), die sowohl als mehr- als auch als einfarbige (dann hochaufgelöste) Figuren ohne Rücksicht auf den gerade eingeschalteten Grafikmodus munter über Text- und Grafikbildschirm huschen können. Die gesamte Verwaltung der Sprites geschieht durch den VIC-Chip, was ihr Auftreten auf den 40-Zeichen-Bildschirm beschränkt. Bevor Sie die folgenden Beispielprogramme ausprobieren, sollten Sie Listing 1 »Alle Sprites« abgetippt und gestartet haben.

Zehn Basic-Befehle dienen zum Erstellen und Verwalten von Sprites:

SPRDEF: Damit ruft man den Sprite-Editor auf. Der Bildschirm wird gelöscht, links oben erscheint ein Raster mit 24 x 21 möglichen Positionen, darunter fragt der Computer nach der gewünschten Sprite-Nummer.

Wir haben nun die Wahl zwischen acht vorgesehenen Sprites. Sollten Sie schon ein MOB im Speicher haben, dann rufen Sie es nun mit seiner Nummer auf, ansonsten müssen Sie sich jetzt festlegen: Alle weiteren Sprite-Befehle beziehen sich im-

Die Befehle zur Spriteprogrammierung

mer auf diese Nummer. Nach deren Eingabe wird das Rasterfeld mit dem Inhalt eines speziellen Speicherbereiches (aus $0E00 bis $0FFF = dezimal 3584 bis 4095) gefüllt, der die Informationen des gewählten Sprites enthält. Falls noch kein Sprite definiert war, kann dabei auch allerlei Byte-Müll abgebildet werden, der uns aber vorläufig nicht erschüttern soll. In der linken oberen Ecke des Rasterbereiches meldet sich ein Kreuz: Das ist der Spritecursor. Oben rechts neben dem Raster ist unser Sprite so abgebildet, wie es im Normalbetrieb später auf dem Bildschirm erscheinen wird.

Bevor wir auf die einzelnen Optionen des Sprite-Editors eingehen, noch eine Bemerkung. Alles bisher beschriebene benutzt den Grafik-Bildschirm. Ebenso wie bei der Verwendung der hochauflösenden und der Mehrfarbengrafik wird dazu der Basic-Anfang nach $4000 verschoben, also über die Bit-Map.

Sehen wir uns nun die Möglichkeiten unseres Editors an:

<CLR/HOME> Damit kann das Definitionsfeld gelöscht werden. Das befreit uns vom Byte-Müll.

<CLR/HOME>: Der Sprite-Cursor marschiert in die linke obere Ecke.

CURSOR-TASTEN

Damit wandert der Sprite-Cursor über das Rasterfeld.

<RETURN>: Der Cursor wird auf den Anfang der nächsten Zeile gesetzt.

<A>: Die Wiederholungsfunktion der Tasten <1> bis <4> kann durch einmaliges Drücken aus-, durch nochmaliges Drücken wieder eingeschaltet werden.

<M>: Einmal Drücken schaltet den Mehrfarbenmodus ein. Dadurch bekommen die Tasten <1> bis <4> eine neue Bedeutung. Nochmaliges Drücken schaltet wieder auf den normalen Modus.

<X>: Ein Druck verdoppelt das Sprite in waagerechter Richtung, ein weiterer Tastendruck erzeugt wieder ein Sprite in normaler Breite.

Y: Vertikale Verdoppelung

ZIFFERNTASTEN <1> BIS <4>

a) Im Normalmodus setzt <1>: die aktuelle Hintergrundfarbe (0),

<2>-<4>: die aktuelle Vordergrundfarbe (1).

b)Im Mehrfarbenmodus setzt

1: die Hintergrundfarbe (00)

2: eine Multicolorfarbe (01)

3: die aktuelle Vordergrundfarbe (10)

4: die andere Multicolorfarbe (11)



Grafik und C128: Das heißt optimales Zusammenspiel von Computer und Anwendung. Wie einfach Shapes und Sprites in der Handhabung sind, werden wir Ihnen auf den nächsten Seiten zeigen.

Zur Erläuterung:

- Im Normalmodus tritt jedes gesetzte Bit (=1) als Bildpunkt in der Vordergrundfarbe auf, alle nicht gesetzten (=0) erscheinen in der Hintergrundfarbe.

- Im Mehrfarbenmodus tragen die Bits paarweise zur Farbgebung bei. Deshalb ist hier der Spritecursor auch doppelt so breit.

<CTRL 1>bis <CTRL 8>

Wählt die aktuelle Vordergrundfarben 1-8 aus.

<CBM 1> bis <CBM 8>

Leistet dasselbe mit den Farbcodes 9 bis 16.

Zwei Funktionen werden merkwürdigerweise im Handbuch nicht erwähnt, die recht nützlich sind:

<C>: Das ist eine Kopierfunktion. Nach Tastendruck erscheint die Frage »Copy from ?«. Nach Eingabe einer Spritenummer wird das dazugehörige Muster in das aktuelle Sprite kopiert und dieses dabei überschrieben. Das ist ganz sinnvoll, wenn man mehrere ähnliche Sprites erzeugen möchte.

Dasselbe passiert auch, wenn man die Funktionstaste F1 betätigt.

Ist man einmal versehentlich auf diese Tasten geraten, dann kann man durch <RETURN> wieder in den normalen Editorzustand gelangen.

<CTRL C>: Erlaubt das Umschalten zwischen den Sprites. Nach Betätigen dieser Tastenkombination verschwindet die aktuelle Spritenummer. Gibt man nun die gewünschte Nummer ein, schaltet sich das dazugehörige Sprite an.

Zwei Möglichkeiten zum Verlassen des Sprite-Editors sind vorgesehen:

RUN/STOP-Taste: Nach dem Betätigen der Taste verliert man alle neuen Eingaben im Definitionsfeld.

Gibt man anschließend eine Spritenummer ein, wird das dazugehörige Muster eingeladen.

Ein anschließendes <RETURN> führt zum Verlassen des Editors. Mit READY meldet sich der Textbildschirm zurück.

<SHIFT RETURN>

Speichern des aktuellen Sprites im Bereich $0E00 - 0FFF.

Was nicht im Handbuch steht

Ebenso wie bei der RUN/STOP-Taste kann man danach durch Eingabe einer neuen Spritenummer das nächste Sprite zur Bearbeitung in den Raster holen.

Ein <RETURN> führt zum Verlassen des Sprite-Editors.

Interessant an SPRDEF ist, daß man diesen Befehl auch im Programm-Modus verwenden kann. Auf diese Weise kann man im Programm ein oder mehrere Sprites erstellen und anschließend - also nach Aussteigen mit STOP/RETURN oder SHIFT-RETURN/RETURN - läuft das Programm weiter.

Das also sind alle Möglichkeiten unseres Sprite-Editors: Leider fehlen einige nötige Optionen. So ist es nicht möglich, die frisch zusammengebauten Sprites in DATA-Zeilen abzulegen, die Multicolorregister müssen vor dem SPRDEF-Aufruf belegt werden, die Muster kann man nicht invertieren, drehen oder spiegeln.

SPRCOLOR

Im Normalmodus kann die Punktfarbe eines Sprites und auch die Farbe der Bitkombination 10 des Multicolormodus durch den SPRITE-Befehl bestimmt werden oder in SPRDEF durch die Kombinationen von CONTROL- oder CBM-Taste mit einer Ziffer.

Die Belegung der Multicolorregister mit Farbe - das entspricht den SPRDEF-Zifferntasten <2> und <4> -, also der Bitkombinationen 01 und 11 erfolgt durch den SPRCOLOR-Befehl. Die Verwendung von

SPRCOLOR A,B

führt in Multicolorregister 1 (Bit-Kombination 01 oder SPRDEF-Taste 2) zur Farbe A, in Multicolorregister 2 (Bit-Kombination 11 oder SPRDEF-Taste 4) zur Farbe B.

Somit ergibt der Befehl

SPRCOLOR 3,6

die Farbe ROT den Bitkombinationen 01 und GRÜN der Kombination 11.

SPRITE

Dies ist zweifellos der wichtigste Befehl für ein schon vorhandenes Spritemuster. Er schaltet ein Sprite ein und legt seine Eigenschaften fest. Syntax:

SPRITE A,B,C,D,E,F,G

Keine Angst vor den vielen Parametern: Sie müssen nicht immer alle angeben. Zur Bedeutung der einzelnen Buchstaben:
A: Spritenummer (1 - 8)
B = 0: Ausschalten des Sprites
B = 1: Einschalten des Sprites
C: Farbcode für die Vordergrundfarbe (das ist das Bitpaar 10 bei Multicolorsprites) (1 bis 16)
D: Priorität gegenüber Text:
0 = Sprite vor Text

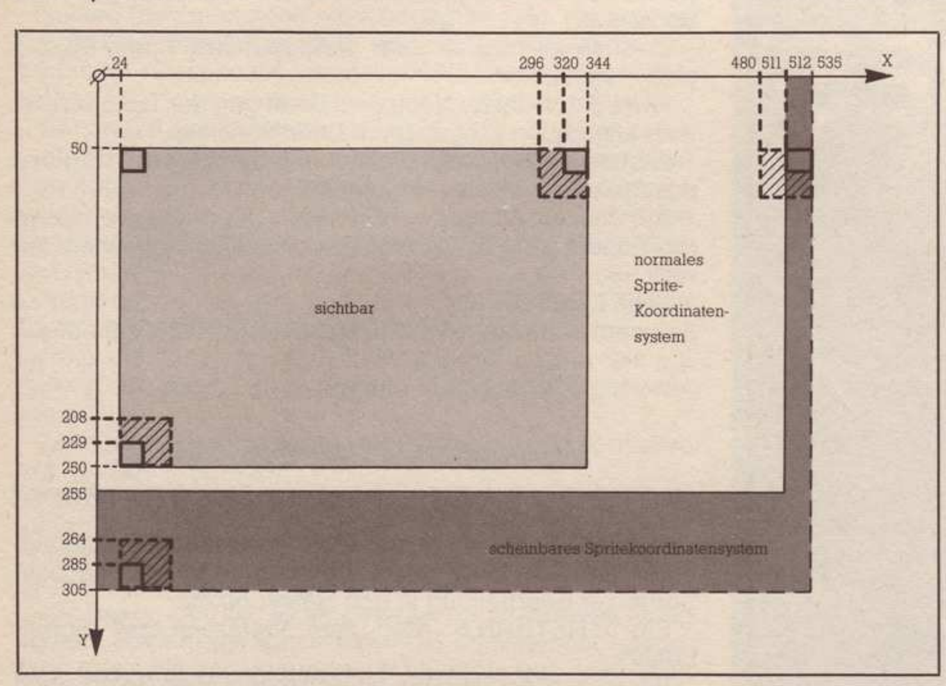


Bild 1. Das normale Sprite-Koordinatensystem

1 = Text vor Sprite

Nebenbei: Die Priorität von Sprites untereinander wird durch die Sprite-Nummer geregelt: 1 vor 2 vor 3 ... vor 8.
E: X-Vergrößerung:
0 = normale Größe
1 = doppelte Größe
F: Y-Vergrößerung:
0 = normale Größe
1 = doppelte Größe

Sind sowohl E als auch F auf 1 gesetzt, dann erhält man ein vierfach ausgedehntes Sprite.
G: Modus.
0 = normales Hires-Sprite
1 = Multicolor-Sprite

Ein Beispiel soll die Wirkung des Befehls demonstrieren:
SPRITE 1,1,3,0,1,1,0

Dadurch wird Sprite 1 eingeschaltet. Es ist rot, erscheint vor Bildschirmzeichen und ist sowohl in horizontaler wie auch in vertikaler Richtung verdoppelt. Es handelt sich um ein normales (also Hires-Sprite). Nun soll das gleiche Sprite auf die normale Größe gebracht werden:
SPRITE 1,,,,0,0

Es genügt, die Werte anzugeben, die zu ändern sind. Alle anderen werden so übernommen, wie sie vorher festgelegt wurden. Die Spritenummer und die Kommas vor dem zu ändernden Wert sind obligatorisch.

MOVSPR

Das ist ein recht vielgestaltiger Geselle. MOVSPR kommt von »move sprite«, also hat dieser Befehl mit der Positionierung und Bewegung von MOBs zu tun. Drei Varianten sind möglich:
a: MOVSPR n,X,Y

Das Sprite mit der Nummer n wird an den Ort mit den Koordinaten X,Y gesetzt. Als Bezugspunkt beim Sprite dient die linke obere Ecke (auch wenn diese unsichtbar ist). Bild 1 zeigt Ihnen das normale Sprite-Koordinatensystem, das in allen Grafik-Modi gültig ist.

Der sichtbare Bildschirm erstreckt sich von X=24 bis 344 und in der Vertikalen reicht er von 50 bis 250. Für X und Y dürfen Werte zwischen 0 und 65535 eingesetzt werden. Bild 1 skizziert auch, bei welchen Koordinatenangaben ein Sprite voll sichtbar ist:

Links oben müssen dazu die Koordinaten (24/50) gewählt werden. Ein normales Sprite ist bei X=320, ein in X-Richtung gedehntes bei X= 296 gerade noch voll sichtbar.

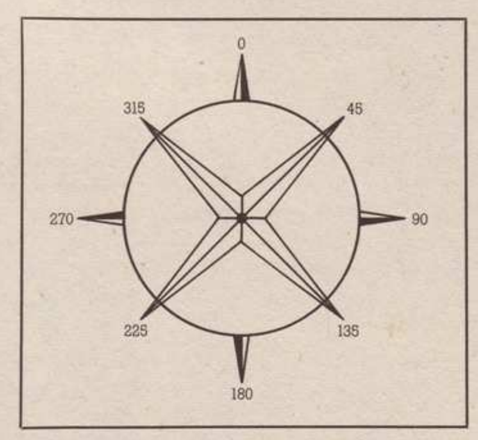


Bild 2. Die Kompaßrose: MOVSPR

Entsprechend gilt für die Y-Werte:
Normales Sprite bis Y=229,
in Y-Richtung gedehntes bis Y=208.

Größere Koordinatenwerte führen zum Herauswandern aus dem sichtbaren Bereich. Überschreiten die X- oder Y-Werte eine bestimmte Grenze, dann treten die MOBs auf der jeweils gegenüberliegenden Seite wieder ins Bild. Das ist ab X=513 (bei gedehnten Sprites ab 481) und ab Y=286 (bei gedehnten Sprites ab 265) der Fall. Eine weitere Erhöhung der Koordinaten führt dann zum Wandern über den Bildschirm, zum erneuten Verschwinden und schließlich Wiederauftauchen auf der anderen Seite und so weiter.

Vorhin war die Rede vom »normalen« Sprite-Koordinatensystem.

Daraus kann messerscharf geschlossen werden, daß es auch noch ein anderes gibt, ein »nicht normales«. Das ist tatsächlich der Fall und es handelt sich nicht nur um eines, sondern um eine ganze Menge verschiedener Systeme. Haben Sie nämlich durch den SCALE-Befehl ein neues Hires- oder Multicolor-Bildschirmsystem definiert, dann beziehen sich X und Y auf dieses, was manchmal zu einiger Verwirrung führen kann. Sollte also die Gefahr bestehen, daß ungewollt aus einer früheren Programmphase ein anderes Bildschirmsystem gültig ist, dann kann durch SCALE 0 der normale Zustand hergestellt werden.

Hier noch ein Beispiel:
MOVSPR 1,300,100
setzt Sprite 1 an die Stelle (300/100) im jeweils gültigen Koordinatensystem.
b: MOVSPR n,+/-X,+/-Y

Damit kann man Sprite n relativ um + oder -X (und/oder Y) zur aktuellen Position verschieben. X und Y dürfen wieder Werte bis 65535 annehmen, wobei durch das Vorzeichen nun sogar eine Skalenbreite von -65535 bis +65535 möglich wird. Auch hier beziehen sich X und Y auf das je-

weils durch SCALE definierte Koordinatensystem (ohne SCALE-Anwendung also auf das normale Sprite-System aus Bild 1).

Auf diese Weise kann - zum Beispiel in einer Schleife - das Sprite praktisch endlos über den Bildschirm wandern. Es tritt nämlich keine Fehlermeldung auf, wenn die Summe aus alter Position und Verschiebung größer als 65535 wird. Beispielsweise ist es ohne weiteres möglich (ob es sinnvoll ist, wäre eine andere Frage), folgende Kombination zu verwenden:

MOVSPR 8,100,65000:MOVSPR 8, 100,+65000

Auch ein Unterlauf unter Null ist auf diese Weise erlaubt.

Wie Sie aus dem Beispiel erkennen können, kann die MOVSPR-Anweisung auch kombiniert verwendet werden:

MOVSPR 1,+10,100 verschiebt Sprite 1 um +10 Einheiten relativ zum vorangegangenen X-Wert und auf die Y-Koordinate 100.

MOVSPR 2,50,-20 läßt Sprite 2 auf die X-Koordinate 50 wandern und gleichzeitig um 20 Y-Einheiten nach oben.

c: MOVSPR n,W # V

Diese Variante des Spritebewegungs-Befehls ist eine feine Sache: Unabhängig vom sonstigen Programmgeschehen wird das Sprite mit der Nummer n mit einem Richtungswinkel W und der Geschwindigkeit V über den Bildschirm gesteuert.

Der Winkel W wird in Grad angegeben. Bild 2 zeigt die möglichen Richtungen, die der bekannten Kompaßrose entsprechen:

Ohne Fehlermeldung sind Eingaben möglich zwischen -65535 und +65535. Sinnvoll scheint das aber allenfalls in einer Schleife, die ständige Richtungswechsel durchführt, zumal negative Werte und Angaben größer als 33024 keine vernünftige Reaktion ergeben.

Auch für die Geschwindigkeit V sind Angaben zwischen -65535 und 65535 möglich. Als sinnvoll erweisen sich hier nur Werte zwischen 0 und 15. 0 stoppt die Bewegung und 15 ist die höchste erreichbare Geschwindigkeit. Für höhere V-Werte ergibt sich die Geschwindigkeit dann zu modulo(16). (Damit ist der Rest gemeint, der bei Division der eingegebenen Zahl durch 16 verbleibt. So ergibt V=30 dieselbe Geschwindigkeit wie V=14)

Eine Eigenart dieser MOVSPR-Variante ist es, daß die Wirkung dieses Befehls anhält, bis die Bewegung ausdrücklich durch MOVSPR n,W # 0 auf Null gesetzt wird. Auch ein abgeschaltetes Sprite wandert weiter, was Sie leicht mal ausprobieren können.

Der Befehl

MOVSPR 1,90#15

läßt Sprite 1 mit Höchstgeschwindigkeit horizontal nach rechts über den Bildschirm jagen.

SPRSAV

Das ist ein sehr interessanter Befehl, dessen Anwendung wir später noch detailliert untersuchen werden. Zwei mögliche Varianten sind vorgesehen:

a: SPRSAV n,A$

Einem String (hier also A$) wird das Bitmuster des Sprite n zugeordnet.

Mittels SPRSAV 1,A$(2)

ordnet man das Bitmuster des Sprite mit der Nummer 1 dem Stringarrayelement A$(2) zu.

b. SPRSAV A$,n

Das ist der umgekehrte Weg: Das in A$ enthaltene Bitmuster definiert nun das Sprite mit der Nummer n.

Durch SPRSAV A$(2),8 wird das in A$(2) gespeicherte Muster in den Sprite Nummer 8 gelesen.

Anstelle von A$ kann jede Stringvariable - auch ein Array-Element - verwendet werden. Einige Möglichkeiten, die auf diese Weise gegeben sind: »Klonen« eines Sprites, Definieren von mehr als acht Sprites (jeweils acht sind gleichzeitig auf dem Bildschirm aktivierbar), schneller Austausch von Spritemustern.

COLLISION

Was im C 64-Modus dem Assembler-Programmierer vorbehalten bleibt oder nur durch aufwendige POKE- und PEEK-Operationen realisiert werden kann, ist im C 128-Modus mit diesem und dem folgenden Befehl möglich:

COLLISION A,NNNN

fängt drei Typen von Ereignissen ab und setzt die Programmbearbeitung in Zeile NNNN fort. A ist dabei eine Typkennung:

1: Sprite/Sprite-Kollision
2: Sprite/Text-Kollision
3: Lichtgriffelaktivierung

Das Serviceprogramm ab NNNN ist wie ein Basic-Unterprogramm zu behandeln, also auch durch <RETURN> abzuschließen. Die weitere Bearbeitung geschieht dann an der Stelle, wo bei dem auslösenden Ereignis unterbrochen worden ist.

COLLISION stellt lediglich fest, welcher Typ stattgefunden hat. Bei Spritezusammenstößen kann also damit nicht die Spritenummer identifiziert werden. Es ist erlaubt, mehrere Kollisionstypen gleichzeitig zu aktivieren. Eine Aktivierung innerhalb eines Unterbrechungs-Service-Unterprogrammes ist nicht möglich: Es findet nur die Bearbeitung eines Typs zur Zeit statt. Falls also mehrere aktiviert sind, sollte einer der ersten Befehle im Unterprogramm das Abschalten aller Kollisionsabfragen sein mittels COLLISION A (ohne Zeilennummer). Am Ende der Routine kann dann COLLISION A,NNNN wieder eingeschaltet werden.

Nicht immer ist es sinnvoll, so zu verfahren (also die Kollisionsabfrage am Ende der Routine wieder einzuschalten): Falls beispielsweise der Typ 1 (Sprite/Sprite-Zusammenstoß) als Auslöser definiert ist, muß bedacht werden, daß die Überlappung zweier Sprites einige Zeit andauert und deshalb das Unterprogramm mehrfach angesteuert wird. Im einem Testprogramm (Listing 2) ist dieser Effekt deutlich an den vielen Textwiederholungen zu erkennen.

Unter welchen Umständen wird eine Kollision erkannt? Immer dann, wenn sich sichtbare Teile von Sprites gegenseitig oder mit Bildschirmobjekten (bei Typ 2) überlagern. Abgeschaltete Sprites werden nicht berücksichtigt, wohl aber Zusammenstöße außerhalb des sichtbaren Bereiches (das Handbuch behauptet das Gegenteil).

Probieren Sie doch einmal, in Zeile 40 von Listing 1 die Spritepositionierung:

40 MOVSPR 7,345,100:MOVSPR 8,345,200

Nach dem Start ist keines der beiden Sprites mehr auf dem Bildschirm zu sehen, Kollisionen werden aber gemeldet.

Zwei Aspekte verdienen noch Erwähnung: Zum einen darf man ein Programm, in dem der COLLISION-Befehl verwendet wird, nicht durch <RUN/STOP> unterbrechen. Tut man es trotzdem, dann funktioniert nach einem CONT die Kollisionsabfrage nicht mehr. Zum anderen: Natürlich können alle Sprite-Befehle auch ohne 40-Zeichen-Bildschirm betrieben werden, man sieht nur nichts. Das kann bei COLLISION ganz rätselhafte Effekte erzeugen. So wären Spiele denkbar, die ein Sprite per Joystick durch Hindernisse hindurchbewegen. Das Ganze geschieht im Dunkel (also mit abgeschalteten oder nicht vorhandenen 40-Zeichen-Bildschirm) und lediglich die Reaktion auf eine Kollision erfolgt auf dem 80-Zeichen-Bildschirm. Der Phantasie sind keine Grenzen gesetzt.

BUMP

Der Name sagt's bereits: Auch hier geht es um Zusammenstöße. Mit dem BUMP-Befehl können wir einfach feststellen, welche Sprites in eine Kollision verwickelt sind:

BUMP(A)
erschließt über die Typkennung A zwei Sorten von Zusammenstößen.
A=1: Sprite/Sprite-Kollision
A=2: Sprite/Text-Kollision.

Der Befehl hat lediglich die Aufgabe, den Inhalt eines speziellen Kollisionsregisters auszulesen. Das allerdings macht er so gründlich, daß dieses Register hinterher gelöscht ist. Es empfiehlt sich daher, diesen Wert sogleich in eine Variable zu speichern:
V = BUMP(1)

Die Zahl, die auf diese Weise erhalten wird, muß allerdings erst entschlüsselt werden (siehe da, ein alter Bekannter aus dem C64-Modus), denn jedem Sprite ist ein Bit zugeordnet: Sprite 1 hängt mit Bit 0, Sprite 2 mit Bit 1 zusammen und so weiter bis zu Sprite 8, welches mit Bit 7 geht. Findet nun die Kollision statt, dann schalten die Bits der kollidierten Sprites auf »1«. Bild 3 zeigt Ihnen diese Zusammenhänge und einige Rechenbeispiele.

Ein Supercrash unter Beteiligung aller acht Sprites würde dann also die Zahl 255 ergeben. Ein kleines Testprogramm (Listing 3) soll das Vorgehen dabei illustrieren.

Besonders die Zeilen 90 und 100 sind sicherlich für Sie interessant, weil man mit ihnen die Nummern der beteiligten Sprites feststellen kann.

Auch hier gilt, was schon beim Befehl COLLISION bemerkt wurde:

Auch Zusammenstöße außerhalb des sichtbaren Bildschirmes werden registriert und das auch ohne 40-Zeichen-Bildschirm. Ebenso wie dort muß beachtet werden, daß ein Zusammenstoß einige Zeit in Anspruch nimmt, also die Reaktion darauf meistens mehrfach erfolgt.

Im Gegensatz allerdings zu COLLISION, das interruptgesteuert abläuft, muß die Abfrage des BUMP-Wertes mehrfach erfolgen. Der Befehl ist sowohl ohne als auch mit COLLISION einsetzbar. Im letzteren Fall dient er als Ergänzung im Unterprogramm zur Feststellung der Spritenummern.

Bits:	7	6	5	4	3	2	1	0	
Bitwerte:	128	64	32	16	8	4	2	1	
Sprites:	8	7	6	5	4	3	2	1	
Beispiele:	0	0	0	0	0	0	1	1	= 3, also Sprites 1 und 2
	0	0	0	0	0	1	0	1	= 5, also Sprites 1 und 3
	0	0	0	0	1	0	0	1	= 9, also Sprites 1 und 4
	1	0	0	0	0	0	0	1	= 129, also Sprites 1 und 8
	1	1	0	0	0	0	0	0	= 192, also Sprites 7 und 8

Bild 3. Das Ergebnis der BUMP(A)-Abfrage

RSPCOLOR

Damit sind Informationen zu den aktuellen Multicolor-Register-Inhalten zu bekommen:
RSPCOLOR(A)
liefert bei
A = 1: Farbcode in Multicolorregister 1 (das ist das, welches der Bitkombination 01 zugeordnet ist)
A = 2: Hier wird der Inhalt des anderen Multicolorregisters ausgegeben, das zu der Bitkombination 11 gehört.

RSPPOS

Sollte uns die aktuelle Position oder Geschwindigkeit eines Sprites interessieren, dann können wir das durch
RSPPOS(n,A)
erfahren. Dabei ist n die Spritenummer und A wieder eine Kennung mit folgender Zuordnung:
A = 0: aktuelle X-Koordinate
A = 1: Y-Koordinate
A = 2: gerade vorhandene Geschwindigkeit

Leider gibt es da eine kleine Unstimmigkeit gegenüber dem Befehl MOVSPR: Gleichgültig, welches Sprite-Koordinatensystem wir dort durch SCALE festgelegt haben, RSPPOS liefert immer die Werte des normalen Systems. RSPPOS(0) liegt daher immer zwischen 0 und 511, RSPPOS(1) immer zwischen 0 und 255. Sollten Sie bei normalem System mittels MOVSPR höhere Werte eingegeben haben, dann wird hier immer modulo(512) für die X- und modulo(256) für die Y-Koordinate ausgegeben.

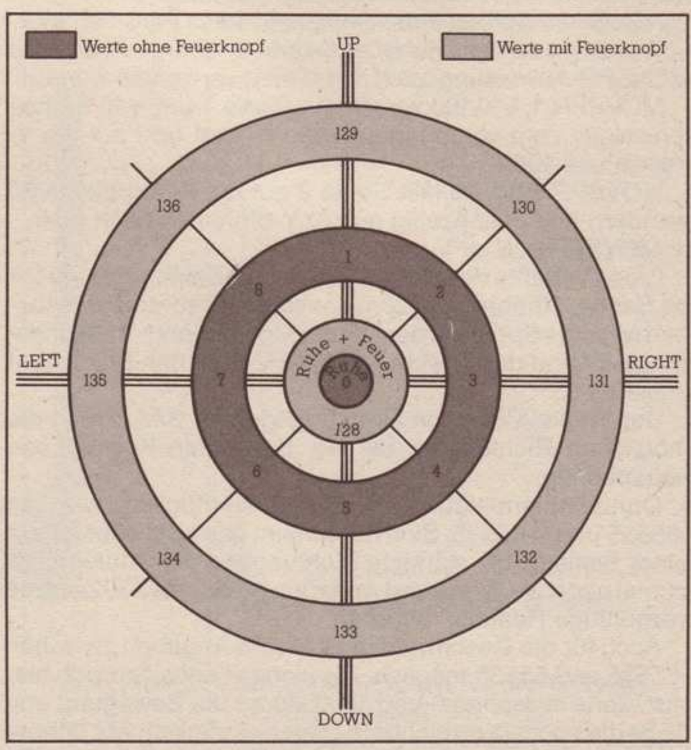


Bild 4. Schema zur JOY-Abfrage

Auch die Sache mit der aktuellen Geschwindigkeit hat einen Haken: Falls Sie nämlich den Sprite mal auf andere Weise als durch MOVSPR n,W#V bewegen, erhalten Sie den Wert 0 als Ausgabe von RSPPOS(2).

RSPRITE

Hier haben wir das Pendant zum SPRITE-Befehl vorliegen. Alle dort verwendeten Parameter können hier durch
RSPRITE(n,A)
abgefragt werden. n ist wieder die Spritenummer, A eine Kennung mit den Zuordnungen:
A = 0: Sprite ein- (=0) oder ausgeschaltet (=1)
A = 1: Farbcode des Sprites (Im Multicolormodus Farbe der Bitkombination 10)
A = 2: Sprite vor (=0) oder hinter (=1) Text.
A = 3: In X-Richtung gedehnt (=1) oder nicht (=0).
A = 4: In Y-Richtung gedehnt (=1) oder nicht (=0).
A = 5: Multicolormodus (=1) oder normales Hires-Sprite (=0)

JOY

Dieser Befehl hat zwar nicht unbedingt etwas mit den Sprites zu tun, wird aber häufig zum Steuern der Sprites eingesetzt und kommt daher an diese Stelle. Man kann damit auf einfache Weise den Zustand der Joystickports abfragen:

JOY(A)

fragt bei A = 1 den Port 1, und bei A = 2 den Port 2 ab.

Bevor wir uns die Bedeutung der Abfrageergebnisse ansehen, noch eine Warnung: Offensichtlich entspricht der Feuerknopf in Port 1 der Taste F8, so daß im Direktmodus immer der Maschinensprachemonitor angesprungen wird, wenn man den Feuerknopf drückt. Beim Port 2 gibt es dieses Problem nicht.

Die mittels JOY ausgelesenen Werte haben folgende Bedeutung:

0 = Ruhestellung
1–8 = Richtungen (siehe Bild 4)
128 = Ruhestellung, Feuerknopf gedrückt
129–136 = Richtungen mit gedrücktem Feuerknopf (siehe Bild 4)

Auf diese Weise können wir bequem Sprites mit dem Joystick steuern:

100 ON JOY(2) GOSUB 300,310,320,330,...

Im Testprogramm (Listing 4) ist eine mögliche Variante dazu gezeigt.

Damit haben wir alle im Basic 7.0 enthaltenen Sprite-Befehle behandelt. Allerdings sind noch einige Fragen offen geblieben, die uns nun weiter beschäftigen werden.

Wie kann man Sprites der Nachwelt erhalten?

1. Sprites im Speicher

a. Sprite-Daten-Speicher

Schon beim SPRDEF-Befehl haben wir diesen Speicherbereich zwischen 3584 und 4095 ($0E00 bis 0FFF) erwähnt, in dem die Spritedaten landen, wenn ein Sprite durch Shift-Return gespeichert wird. Pro MOB befinden sich dort – in der Reihenfolge der Spritenummern – je 63 Byte und eine Endmarkierung (das ist ein 0-Byte). Verfügbar bleiben diese Daten, bis sie überschrieben oder der Computer abgeschaltet wird.

b. Sprite-Strings

Noch flüchtiger ist die Existenz der Daten im String, der durch SPRSAV gebildet wird. Schon ein CLR macht der Kunst den Garaus. In jeweils 67 Byte (die letzten vier enthalten immer die Werte 23, 0, 20, 0) finden sich dort die Bitmuster. Sollten Sie einmal versuchen, so einen String (z.B. A$) durch PRINT A$ auf dem Bildschirm zu zeigen, dann müssen Sie sich auf allerhand Überraschungen gefaßt machen: Alle Bytewerte von 0 bis 255 können auftreten und wirken so, als wären sie mittels CHR$(Wert) zur Ausgabe aufgerufen worden. Wenn also der Wert 147 zufällig dabei sein sollte, dann wird der Bildschirm gelöscht, etc. Jedes RUN führt übrigens ebenfalls ein CLR aus. Auch das ist daher nicht die richtige Methode, die mühselig konstruierten Sprites etwas länger am Leben zu erhalten.

Sowohl im Speicher als auch im String haben die Sprite-Daten genau dasselbe Format (bis auf die letzten Bytes, die im einen Fall aus einer 0, im anderen aus den Zahlen 23,0,20,0 bestehen). Das können Sie selbst nachprüfen am Programm "VERGLEICH SPRDT" (Listing 5).

Nach dem Start werden zunächst die Spritedaten von der Diskette geladen und dann in Strings eingelesen. Anschließend erscheinen auf dem Bildschirm Sprite für Sprite die Bytewerte sowohl aus dem Speicher als auch aus dem String.

2. Sprites auf Diskette und Kassette.

a. BSAVE

Eine elegante Methode zur Verewigung unserer Sprites ist das Speichern des Spritespeichers mittels des BSAVE-Befehls. Man schreibt dazu:

BSAVE"Name",ON B0,P3584 TO P4095

Damit sind dann alle acht Sprites erfaßt. Leider arbeitet dieser Befehl nicht mit der Kassettenstation, denn es sind nur Gerätenummern von 4 bis 15 zulässig. Das Wiedereinladen geschieht dann (wie in den hier vorgestellten Beispielprogrammen) mittels:

BLOAD"Name",ON B0

b. Vom Monitor aus

Damit ist es nun auch Benutzern der Datasette möglich, den Spritespeicher auf Kassette zu sichern. Der Monitor verfügt über ein Kommando S zum Speichern beliebiger Speicherbereiche. Mittels MONITOR oder der Funktionstaste F8 schalten Sie den Monitor an, dann verwenden Sie:

S"Name",01,+3584,+4095
(Kassette) oder
S"Name",08,+3584,+4995
(Diskette).

Mit Hilfe des L-Kommandos im Monitor kann solch ein File dann problemlos wieder geladen werden:

L"Name",01(oder 08),+3584

c. Als sequentielles File

Hat man die Sprites in Strings abgelegt (das dürfen dann auch mehr als acht sein), dann kann man sich der üblichen Techniken zur Speicherung in sequentiellen Dateien bedienen. Allerdings kann es manchmal dabei Schwierigkeiten mit bestimmten Byte-Inhalten geben. So ist es durchaus möglich, daß ein Byte zufällig den Inhalt 13 (also einem <RETURN> entsprechend) hat, was zu Störungen beim Wiedereinlesen führen kann. In solchen Fällen könnte man die einzelnen Bytes (wie im Programm »VERGLEICH SPRDT« geschehen) in die ASCII-Zahlen wandeln und in dieser Form als SEQ-File speichern. Diese Möglichkeit werden wir nicht weiter beschreiben, weil die Speicherung per Monitor oder BSAVE schneller und effektiver erscheint.

3. Für ein Listing

Viele Programme werden nicht in Form von Disketten- oder Kassettenfiles, sondern einfach auf dem Papier weitergegeben. Dieses Problem lösen bessere Sprite-Editoren durch eine Funktion, die alle Werte in DATA-Zeilen an ein Programmende anhängt. In SPRDEF existiert diese Möglichkeit leider nicht, was uns dazu zwingt, selbst die Initiative zu übernehmen.

a. Aus dem Speicher in DATAs

Dazu haben wir ein kleines Programm »SPRITEDATAS« gestrickt (Listing 6), das Sie mit der MERGE-Funktion an Ihr eigenes fertiges Spriteprogramm anhängen können. Noch eine Warnung: Wenn Sie SPRITEDATAS abgetippt haben, speichern Sie es unbedingt vor einem RUN ab, denn es verabschiedet sich am Ende des Programmlaufes aus Ihrem Basicspeicher. Nun also zur Verwendung des Programmes. Man startet es durch RUN63000. Das Auslesen des Sprite-Datenspeichers geschieht im FAST-Modus (der 40-Zeichen-Bildschirm verabschiedet sich vorübergehend), danach erscheinen jeweils zwei Programmzeilen (mit Zeilennummern ab 63020) und darunter ein GOTO 63009. Außerdem meldet sich ein BREAK und READY. Mittels der CLR/HOME-Taste und 3 aufeinanderfolgenden <RETURN> übernehmen Sie die Zeilen ins Programm und erzeugen den nächsten Bildschirm, wo das Spielchen

dann ebenso weitergeht. Insgesamt machen wir das achtmal (für 8 Sprites). Zu guter Letzt wird in Zeile 63040 noch eine Einleseschleife übernommen und es meldet sich der Befehl DELETE 63000-63011. Haben Sie auch das mit <CLR/HOME> und zwei <RETURN> übernommen, dann ist der DATA-Generator gelöscht und an Ihr Programm wurde die gesamte DATA-Sequenz angehängt. Als letzte Zeile finden Sie beim Listen noch die Einleseschleife, zu der Sie nun nur noch an passender Stelle Ihres Programmes mittels GOSUB 63040 springen müssen.

Es gibt sicherlich elegantere Möglichkeiten, DATA-Zeilen zu generieren. So könnten wir auch den Tastaturpuffer verwenden. Das aber überlassen wir Ihrer Schöpferkraft.

b. Aus den Strings in DATAs

Auch das Übertragen der Stringinhalte in DATA-Zeilen ist eine Lösungsmöglichkeit unseres Problems. Sie ist sogar sehr verlockend, weil wir damit auf einen Streich mehr als acht Sprites ins Listing schreiben könnten. Die Programmierung wird aber etwas komplizierter, weil hier nicht mehr der ganze Inhalt des Sprite-Strings in einen Zeilen-String gelesen werden kann. Es gibt eine Reihe von Strategien um das Problem zu lösen (mehr oder weniger einfache, was teilweise mit dem Auslesen der DATAs zusammenhängt), auch hier sind Sie gefordert!

Das Prinzip ist einfach: Weil jedes Sprite in X-Richtung 24 Bit breit ist, ergibt sich für die X-Koordinate des Nachbarsprites bei nahtloser Verbindung ein um 24 größerer Wert.

Koppeln von Sprites

Sollten Sie ein komplexeres grafisches Gebilde erzeugen wollen als es mit einem Sprite möglich ist, dann können Sie auch mehrere Sprites koppeln. Am Beispiel von vier Sprites zeigt Ihnen Bild 5 die dazu nötigen Koordinatenwerte:

21 Bit mißt ein MOB in der Vertikalen, wodurch sich Y+21 für ein unterhalb des ersten gelegenes Sprite berechnet als Y-Koordinate. Das Programm »VIER SPRITES« (Listing 7) demonstriert die Verhältnisse.

Zunächst wächst ein grafisches Objekt aus vier Sprites zusammen. Dieses kann dann mit dem Joystick in Port 2 (nach Druck auf den Feuerknopf) über den Bildschirm gesteuert werden (Zeile 300). Ein weiterer Druck auf den Feuerknopf beendet das Programm. Wenn Sie bei bestimmten Richtungen genau hinsehen, werden Sie eine leichte Unstimmigkeit in der Bewegung feststellen. Bei noch größeren Objekten scheint Basic für diese Steuerung doch etwas zu langsam zu werden und man muß auf Maschinensprache umsteigen um eine flüssige Bewegung zu erreichen.

Mehr als acht Sprites

Hier soll nicht die Rede sein von trickreichen Anwendungen der Unterbrechungsprogrammierung per Assembler, die es tatsächlich zuläßt, mehr als acht Sprites GLEICHZEITIG auf dem Bildschirm zu zeigen. Vielmehr geht es uns um die schnelle, nacheinander ausgeführte Abbildung. Sehen Sie sich dazu zuerst einmal das Programm »SPRITE-TRICK« (Listing 8) an. Hier liegen acht Spritemuster vor, die nacheinander - mit programmierter Verzögerung, weil's sonst zu schnell ginge – auf derselben Bildschirmstelle gezeigt werden. Es ergibt sich ein kleiner Trickfilm.

Nun sind acht Teilbilder ein etwas ärmlicher Bewegungseffekt. Um wirklich längere Passagen zu zeigen, müßten wir anders verfahren. Das soll im Programm »24 SPRITES« Listing 9 demonstriert werden.

Das Geheimnis – Sie haben es sicherlich schon längst vermutet – liegt im SPRSAV-Befehl begründet. 24 Spritemuster wurden hier in ein Stringarray eingelesen (Zeilen 20 bis 80). Während die Sprites auf dem Bildschirm aktiv sind (die Aktivierung erfolgt schon in den Zeilen 130 bis 150), wechseln wir durch die Zuordnung anderer Spritemuster aus den Strings ihre Erscheinungsformen. Das geschieht in der DO...LOOP Schleife ab Zeile 170. Hier geschieht das – wegen des hübschen Mustereffektes – zyklisch (durch Drücken von <←> können Sie das Programm beenden), man kann sich aber ohne weiteres vorstellen, wie ein oder mehrere große Arrays gebildet und dann fortlaufend deren Spritemuster verwendet werden können. Gleichzeitig sind die sich ändernden Sprites noch beweglich...

Sie sehen, daß kaum Grenzen gesetzt scheinen, allerhand Interessantes auf dem Bildschirm ablaufen zu lassen.

Noch ein paar Sprite-Besonderheiten

1. Sprites und Splitscreens

Beim C128 haben wir ja in den Grafik-Modi 2 und 4 die Möglichkeit, sowohl hochauflösende (oder aber Multicolor-) Grafik als auch Text auf einem Bildschirm gleichzeitig darzustellen. Damit ist nicht etwa der CHAR-Befehl gemeint, sondern durch eine spezielle Technik, die den Ra-

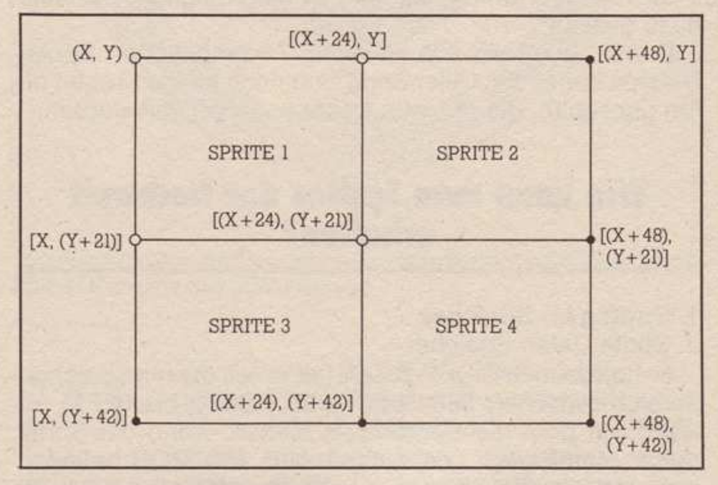


Bild 5. Koppeln von vier Sprites

sterzeileninterrupt ausnutzt, wird der Bildschirm tatsächlich in zwei verschiedenen Modi betrieben.

Wie verhalten sich nun die Sprites auf solch einem Bildschirm? Wie es Kobolden ansteht, scheren sie sich überhaupt nicht darum! Probieren Sie mal das Programm VIER SPRITES mit einer kleinen Änderung aus: In die Zeile 130 fügen Sie noch ein GRAPHIC 2,0,12 (oder für den Multicolormodus GRAPHIC 4,0,12) und starten dann. Ohne Reaktion von seiten der Sprites können Sie diese über die in der Bildmitte liegende Modusgrenze hinwegsteuern. Allerdings reagiert der Bildschirm mit einem penetranten Flattern der Grenzlinie, wenn das Sprite genau darauf liegt.

Falls Sie dasselbe mit dem Grafik-Modus 4 (also GRAPHIC 4,0,12) probieren, würde es uns interessieren, ob auch Ihr Computer nach einiger Zeit mit irgendeiner Fehlermeldung aussteigt. Beim Modell des Autors hat er sich so verhalten. Wenn er sich dann die als fehlerhaft gemeldete Zeile listen ließ, befanden sich allerlei merkwürdige – anscheinend zufällig erzeugte – Fehler darin. Zuvor aber war die Zeile fehlerfrei gewesen! Man sollte offensichtlich diese Möglichkeit mit etwas Vorsicht gebrauchen.

2. Sprites und Windows.

Ebensowenig wie sich Sprites um den Splitscreen küm-

mern, scheren sie sich um Windows. Probieren Sie es mal aus, indem Sie in Zeile 130 des Programmes »VIER SPRITES« noch ein Window definieren (zum Beispiel mit WINDOW 5,5,20,10). Unbeirrt von allen Einschränkungen, denen unser Text unterworfen ist (hier die Aufforderung zum Benutzen des Port 2), läßt sich unser Spritegebilde kreuz und quer verschieben.

Die Sprites an sich sind damit zunächst einmal durchleuchtet. Legen wir sie also für eine Weile beiseite und wenden wir uns den anderen Grafikobjekten des C128 zu, den Shapes.

Was sind Shapes

Wie so vieles aus der Computerkultur stammt auch dieses Wort aus dem angelsächsischen: Shape heißt ins Deutsche übersetzt soviel wie Form, Gestalt, Umriß. Hier bezeichnet Shape einen genau definierten Bildschirmausschnitt, der gespeichert und wiederverwendet werden kann. Wie das mit dem C128 geschieht, soll nun erklärt werden.

SSHAPE

Dies ist der Befehl, mit dem ein Bildschirmbereich in einen String eingelesen werden kann. Seine Syntax ist
SSHAPE A$,X1,Y1,X2,Y2

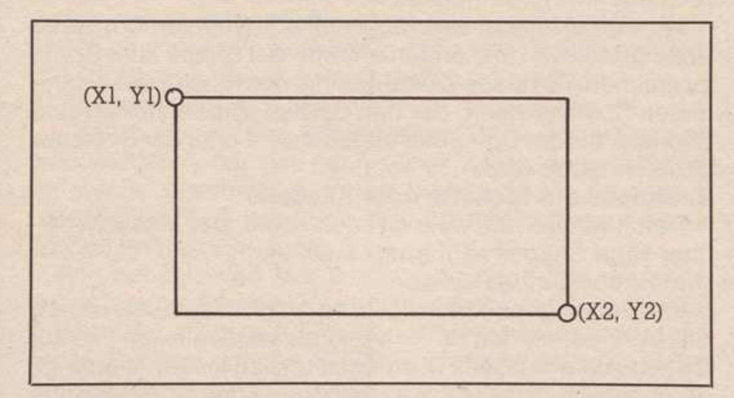


Bild 6. Zur Syntax des SSHAPE-Befehls

A$ ist hier dann eine beliebige Stringvariable, auch ein Array kann Verwendung finden. X1 bis Y2 bezeichnet die Eckkoordinaten des Rechteckes, dessen Inhalt als Shape definiert werden soll. Bild 6 soll das etwas verdeutlichen:

(X1,Y1) gehören zum linken oberen und (X2,Y2) zum rechten unteren Eckpunkt. Es ist auch möglich, X2 und Y2 wegzulassen. In diesem Fall werden durch den SSHAPE-Befehl einfach die aktuellen Koordinaten des Grafik-Cursors eingesetzt.

Falls es noch unklar sein sollte: Im Gegensatz zu Sprites, die jedem Bildschirm-Modus trotzen (also sowohl als Hires- oder Multicolorsprites im Text-, als auch im Hochauflösungs- und im Multicolormodus auftauchen können), sind Shapes fest mit dem aktuellen Modus verbunden.

Ein Shape ist ein Grafikobjekt, kann daher auch nur in den Grafikmodi auftreten und definiert werden. Also in denen, die durch GRAPHICn erzeugt werden, wobei n von 1 bis 4 geht. Außerdem ist ein Multicolorshape auch nur im Multicolormodus ein solches. Schaltet man beispielsweise mittels GRAPHIC1 um in den normalen Hochauflösungsmodus, wird auch unser Shape nur als hochaufgelöstes sichtbar.

Wie wir uns aus all dem schon fast denken können, ist das Koordinatensystem, auf das sich die Angaben X1 bis Y2 beziehen, das im jeweiligen Modus gültige. Im Normalfall also im Hires-Modus X von 0 bis 319 und Y von 0 bis 199. Im Multicolormodus geht X nur von 0 bis 159.

A$ ist ein String. Für Strings gilt beim C128 die Begrenzung auf maximal 255 Byte. Ein durch SSHAPE in einen String zu schreibendes Gebiet darf also eine gewisse Größe nicht überschreiten. Bezeichnen wir als
DX = X2 - X1
und als
DY = Y2 - Y1
dann gilt für die Bytemenge im String im Hochauflösungsmodus die Formel:
Lh=INT((ABS(DX)+1)/8+.99)*(ABS(DY)+1)+4

Im Multicolormodus gilt statt dessen die Gleichung:
Lm=INT((ABS(DX)+1)/4+.99)*(ABS(DY)+1)+4

Die Addition von 4 am Ende der Formeln ergibt sich durch vier Byte, die den Schluß des Shape-String bilden und Steuergrößen enthalten.

Natürlich können Sie jedesmal, wenn Sie ein Shape erstellen, von dem Sie argwöhnen, es könne zu groß sein für den String, unsere Formeln anwenden. Eine Hilfe soll Ihnen Listing 10 »SSHAPE-TABELLE« geben. In der vorliegenden Form druckt es eine Liste aus, in deren linker Spalte Werte für eine Y-Differenz stehen. Rechts daneben sind die maximal zulässigen DX-Werte angegeben sowie die sich ergebende Stringlänge. Auf diese Weise können Sie - ausgehend von DY - mit einem Blick erkennen, wie groß DX sein darf. Es empfiehlt sich, noch drei weitere Tabellen auszudrucken. Da ist zunächst einmal interessant, wie der Maximalwert von DY bei gegebenem DX aussieht. Zu diesem Zweck ist lediglich die Reihenfolge des Ausdruckes und der Rechenoperationen zu ändern:

In Zeile 20 kommt dann X-DIFF vor Y-DIFF, Zeile 40 enthält X=0, in Zeile 50 schreiben wir DO UNTIL X=320, in der nächsten Zeile 60 steht: X=X+1:Y=0. Zeile 70 wird zu DO UNTIL Y=200 und Zeile 80 zu Y=Y+1. Schließlich wird auch in Zeile 90 noch die Reihenfolge des Ausdruckes umgestellt zu PRINT #1,X1,Y1,Z.

Diese beiden Tabellen beziehen sich auf Shapes im Hochauflösungsmodus. Zwei weitere Tabellen für den Multicolormodus bekommen Sie, wenn Sie in Zeile 30 die vorhin angegebene Berechnungsfunktion zu diesem Modus einfügen. Nur die Zahl 8 muß dazu in eine 4 verändert werden.

Sollte dann, trotz all dieser Berechnungsmöglichkeiten - man sieht ja nicht jedesmal in die Tabellen - der zu speichernde Shape größer werden als 255 Byte, dann meldet unser Computer einen STRING TOO LONG ERROR.

Einen Aspekt zum SSHAPE-Befehl müssen wir noch besprechen, weil es da einige Verwirrung geben kann: Wie reagiert SSHAPE auf ein durch SCALE verändertes Bildschirmsystem? Die Koordinatenangabe bei SSHAPE muß dann ebenfalls in den aktuell gültigen Werten geschehen. Man muß sich also nicht um die Organisation des Koordinatensystems kümmern, denn durch SSHAPE wird automatisch die Umrechnung auf den normalen Bildschirm vorgenommen. Lediglich bei der Berechnung der Stringlänge müssen einige Korrekturen eingefügt werden.

Nehmen wir an, wir hätten durch SCALE 1,XM,YM ein System definiert, dann liegen (im Hochauflösungsmodus) die Vergrößerungsfaktoren
FX = XM/320
FY = YM/200
vor und die Beziehungen für die Stringlänge lauten dann:
Hochauflösungsmodus:
Lh=INT((ABS(DX*320/XM)+1)/8+.99)*
(ABS(DY*200/YM)+1)+4
Multicolormodus:
Lm=INT((ABS(DX*160/XM)+1)/4+.99)*
(ABS(DY*200/YM)+1)+4

DX und DY sind dabei im aktuellen Koordinatensystem geltende Werte.

Begnügen wir uns nun mit dem SSHAPE-Befehl (es gäbe noch einiges zu untersuchen) und wenden wir uns dem Gegenstück, nämlich dem GSHAPE-Befehl zu.

GSHAPE

Ein durch SSHAPE im String gespeichertes Bild kann durch diesen Befehl nun wieder auf dem Bildschirm dargestellt werden:

GSHAPE A$,X,Y,M

A$ ist der uns nun schon bekannte Speicherstring, der durch SSHAPE definiert wurde. Es kann jede Stringvariable oder ein String-Array-Element verwendet werden.

X,Y sind die Koordinaten im aktuellen System, an die die linke obere Ecke unseres Shapes gelegt werden soll.

Interessant ist die Angabe von M. Hier dreht es sich um einen Darstellungsmodus, von dem es hier vier Möglichkeiten gibt:

M = 0: Unser Shape wird genauso abgebildet, wie es definiert wurde. Schon an dieser Stelle auf dem Bildschirm vorhandene Objekte werden überdeckt. Dasselbe ergibt sich, wenn wir diesen Parameter einfach weglassen.

M = 1: Das Shape wird invertiert abgebildet. Auch hier überdeckt das Shape vorhandene Objekte.

M = 2: OR-Modus: Bildpunkte werden sichtbar, wenn sie zum Shape oder zu vorhandenen Objekten gehören. Beide Bilder überlagern sich so.

M = 3: AND-Modus:Bildpunkte werden nur dann gesetzt, wenn sie sowohl zum Shape als auch zum Bildschirmobjekt gehören.

M = 4: EOR- (oder XOR-) Modus: Bildpunkte werden nur dann gesetzt, wenn das entsprechende Bit für Shape und Bildschirmobjekt ungleich ist.

Die Modi 2, 3, 4 sind etwas schwer vorstellbar. Zur Verdeutlichung finden Sie hier ein kleines Demonstrationsprogramm namens »GSHAPE-MODI« (Listing 10). Hier wird zuerst ein Shape definiert, dann der gesamte Bildschirm quergestreift und schließlich das soeben erstellte Shape in allen fünf Modi darauf dargestellt. Die Wirkung ist auf diese Weise ganz gut zu erkennen, besonders bei den Modi 2, 3 und 4.

Der Modus 4 ist aufgrund der EOR-Behandlung interessant: Wird nämlich auf dieselbe Stelle noch einmal das Shape im Modus 4 gezeichnet, dann verschwindet es ganz und der Bildschirmhintergrund ist wieder vorhanden. Will man Shapes bewegen, ohne Bildschirmobjekte zu zerstören, dann ist diese Darstellungsweise ganz gut dazu geeignet.

Die Frage, wie viele Shapes man definieren und auch darstellen könne, erübrigt sich fast: Beliebig viele, solange der Speicherplatz für die Strings reicht. Und davon haben wir eine ganze Menge! Man könnte sich also eine ganze Bibliothek von Hochauflösungs- oder Multicolorobjekten zulegen und diese dann bei Bedarf abrufen. Sollte für ein Objekt der String zu kurz sein, können – ebenso wie wir es bei den Sprites schon praktiziert haben – mehrere Shapes gekoppelt werden. Sie erkennen schon: Auch hier sind die Möglichkeiten sehr breit.

Zusammenspiel von SSHAPE und GSHAPE.

Eine Frage haben wir noch offengelassen, die eine recht verlockende Konsequenz in sich birgt: Wenn man mittels der SCALE-Anweisung die Systeme bei SSHAPE (also beim Aufbau des Shape) und bei GSHAPE (bei der Abbildung) unterschiedlich wählt, kann man dann Shapes verkleinern oder vergrößern?

Die Antwort ist leider nein. Bei GSHAPE werden nur die Koordinaten eines (des linken oberen) Eckpunktes angegeben. Lediglich auf den Darstellungsort hat somit ein unterschiedliches Koordinatensystem eine Wirkung, nicht aber auf die Shape-Größe.

Bewegen von Shapes

Im Gegensatz zu den Sprites ist das Bewegen von Shapes eine langweilige Angelegenheit. Jedem Shape-Aufbau durch GSHAPE kann man ganz geruhsam zusehen. Zwar ist es durch Sequenzen wie:

```
FOR I = 0 TO 200
    GSHAPE A$,I,I,4
    GSHAPE A$,I,I,4
NEXT I
```

möglich, Shapes ohne Schaden für den Bildschirminhalt über den Sichtbereich ziehen zu lassen. Das Ganze ähnelt aber bei weitem nicht der Sprite-Bewegung. Aus diesem Grund ist es anzuraten, bei der Bewegung von Bildschirmobjekten – wann immer möglich – Sprites zu wählen.

Shapes der Nachwelt erhalten

Wir stehen bei den Shapes demselben Problem gegenüber wie bei den Sprites. Auf welche Weise kann ein einmal erstelltes Shape für weiteren Gebrauch beiseitegelegt werden?

1. Shapes im Speicher

a. Der Shape-String

Durch SSHAPE wurde ein String geschaffen, der die gleiche Lebensdauer hat wie jede andere Variable: RUN und CLR löschen ihn.

b. Shape im Sprite-Speicher

Wir werden später eine Möglichkeit kennenlernen, die es unter gewissen Umständen erlaubt, ein Shape zum Sprite zu machen. Auf diese Weise gelangt das Muster dazu dann in den Speicherraum, der den Sprites vorbehalten ist und überlebt, bis der Computer abgeschaltet oder der Speicher überschrieben wird.

2. Shapes auf Diskette oder Kassette

Damit wird es nun wirklich interessant. Der Massenspeicher kann Shapes »auf ewig« festhalten.

a. Shape-Strings direkt

Im Grunde genommen sollte es möglich sein, den durch SSHAPE definierten String direkt als sequentiellen File auf Diskette oder Kassette abzulegen und zu lesen. Allerdings ergeben sich hier – wie auch schon bei den Sprite-Strings – Schwierigkeiten beim Lesen. Weil im Prinzip alle Bytewerte von 0 bis 255 auftreten können, also auch ein Wert, der einem Carriage Return oder einem Anführungszeichen entspricht, wird der String nicht immer vollständig und fehlerfrei gelesen. Dazu kommt das Problem, daß die Shape-Strings unterschiedliche Längen aufweisen können. Wie bestimmt man das Ende eines solchen Files?

b. Als ASCII-Werte speichern

Ebenfalls bei den Sprites wurde diese Möglichkeit schon angedeutet: den Shape-String auseinanderzunehmen und jedes Byte in Form seines ASCII-Wertes zu speichern. Eine Variante dieses Verfahrens zeigt Ihnen das Listing 12 »SHAPE SPEICHERN«. Hier wird zuerst ein Shape (einfach ein Kreis) erzeugt und in A$ abgelegt. Ein Unterprogramm druckt auf dem Textbildschirm (für Benutzer von 2 Bildschirmen sofort, für andere erst später sichtbar) die ASCII-Werte des Strings aus. In Zeile 40 wird der alte File SHAPE1 gelöscht und dann ab Zeile 50 der neue auf Diskette abgelegt. Benutzen Sie eine Datasette, dann sind lediglich in den Zeilen 60, 80, 120 und 180 die DOPEN und DCLOSE gegen OPEN und CLOSE mit den entsprechenden Gerätenummern auszutauschen. Nun sind wir eigentlich schon fertig: In Zeile 100 löschen wir alle Variablen, den Grafikbildschirm und starten den zweiten Teil des Programmes, der nur zur Kontrolle den File einliest und unser Shape abbildet. Zusätzlich werden auf dem Textbildschirm wieder die Stringlänge und die eingelesenen ASCII-Werte abgebildet. Beide Ausdrucke lassen sich gut vergleichen (Benut-

zer eines Bildschirmes kommen in diesen Genuß, wenn sie nach Ende des Programmes durch GRAPHIC0 den Textbildschirm einschalten).

c. Eine unorthodoxe Lösung

Etwas exotisch und daher hier nur kurz skizziert ist die folgende Vorgehensweise: Mittels des POINTER-Befehls (hier also POINTER(A$)) sucht man in BANK 1 den Stringdescriptor. Die dort genannte Adresse und Länge verhilft uns dazu, den Speicherbereich, in dem wir unseren String stehen haben, durch BSAVE abzuspeichern.

Beim Wiedereinladen müßte man zunächst einen String definieren, dann durch BLOAD dessen Inhalt an eine beliebige Speicherstelle packen und den Stringdescriptor darauf richten.

Das erscheint im ersten Moment ziemlich kompliziert zu sein, könnte aber – besonders bei Verwendung vieler Shapes – schneller funktionieren. Damit würde beispielsweise ein festgelegter Speicherbereich für alle Shapes en bloc ein- oder auslesbar werden. Probieren Sie's doch mal!

Shapes für ein Listing

a. Als Zeichenvorschrift

Häufig werden Shapes durch einige wenige Grafik-Befehle erstellt. Die einfachste Methode – der wir uns auch bisher bedient haben in den Beispielprogrammen – ist es daher, die Zeichenvorschrift ins Programm einzuarbeiten. Manchmal stößt man damit aber an Grenzen: Wenn beispielsweise der Ausschnitt eines Multicolorbildes als Shape verwendet werden soll, das auf dem Grafiktablett erstellt wurde oder wenn es sich um ein Teil eines Fractals handelt, das 25 Stunden bis zur Fertigstellung dauert etc.

b. Shapes in DATA-Zeilen

Ähnlich wie wir bei Sprites einen DATA-Generator verwendet haben, der sich nach seiner Verwendung selbsttätig löscht, geschieht das nun hier auch. Im Programm »SHAPE DATA« (Listing 13) ist außerdem noch ein kleiner Testteil enthalten.

Wie gehabt, wird hier zuerst ein Shape (ein Kreis) erzeugt, dann endet das Programm an einem STOP in Zeile 6. Wenn Sie mit dem 40-Zeichen-Bildschirm arbeiten, dann fügen Sie bitte vor das STOP-Kommando noch ein GRAPHIC 0 ein, denn alles, was nun folgt, ist Text. Das STOP-Kommando wurde eingebaut, um Ihnen die Möglichkeit zu geben, an dem Shape vor dem Aufruf des DATA-Generators noch etwas zu verändern. Ist alles in Ordnung, dann geben Sie bitte CONT ein. Der DATA-Generator erzeugt nun acht Zeilen mit DATAs und dazu noch das Einleseunterprogramm sowie die Löschanweisung DELETE 63000-63012.

Die erste Zahl in den DATAs ist die Stringlänge, die als A später beim Auslesen und Zusammenfügen des Shape-Strings eine wichtige Rolle spielt. Ist das Programm mit dem Ausdrucken der neuen Zeilen fertig, dann bricht es ab. Durch die HOME-Taste gelangen Sie in die erste Zeile. Mittels mehrfach wiederholtem RETURN können nun alle neuen Zeilen ins Programm übernommen und der Generator gelöscht werden. Hinterher sieht unser Programm dann aus wie in »SHAPE DATA 2« (Listing 13). Zur Kontrolle lassen Sie doch mal dieses Programm durch RUN 10 starten. RUN löscht ja auch die Strings, so daß das nunmehr gezeichnete Shape aus den DATA-Zeilen wiederauferstanden ist.

Die Vorgehensweise bei diesem Programm ist dieselbe wie beim DATA-Generator für die Sprites. Allerdings werden hier die Werte nicht aus dem Speicher, sondern aus einem String geholt, der deshalb auch schon durch einen – zumindest teilweisen – Programmlauf definiert worden sein muß.

Wenn Sie mehrere Shapes auf diese Weise festhalten wollen, ist das Programm relativ einfach umzubauen, am besten mittels eines String-Arrays und einer weiteren Schleife.

Wenden wir uns nun dem Zusammenspiel von Sprites und Shapes zu:

Erzeugen von Sprites aus Shapes.

Anstatt mit dem SPRDEF-Befehl können Sprites auch regelrecht gezeichnet werden durch Grafik-Befehle im Hochauflösungs- oder im Mehrfarbenmodus. Die Nahtstelle zwischen Sprite und Shape ist dabei der String, in den unsere Zeichnung zuerst als ein Shape (mittels SSHAPE) eingeschrieben wird. Von dort ist es dann durch SPRSAV in den Sprite-Speicher zu übertragen.

Shapes und Sprites

Eine Regel ist dabei zu beachten: Der Sprite-String liegt in seiner Länge fest. Im Normalmodus müssen wir immer von einem 24 mal 21 (das ist X mal Y) Raster ausgehen, im Mehrfarbenmodus vom 12 x 21-Muster. Eine SSHAPE-Anweisung muß daher – falls wir ein Sprite konstruieren wollen – im Normalmodus lauten:

SSHAPE A$,X,Y,X+23,Y+20 und im Multicolormodus:

SSHAPE A$,X,Y,X+11,Y+20

X und Y sollen die Koordinaten der linken oberen Ecke des gewünschten Bildausschnittes sein.

Ist SSHAPE nicht in diesem Raster erstellt worden, kann sich allerlei Unsinn auf dem Bildschirm abspielen.

Ein kleines Demoprogramm (Listing 13) namens »SHAPE UND SPRITE« spielt diese Option durch. Zunächst wird ein Multicolorshape erzeugt und in Sprite 1 geschrieben. Eine Umstellung vom Multicolor- in den Hochauflösungsmodus zeigt, daß das Sprite seine Farben behält, das Shape dagegen nicht. Ein zweites Shape erzeugt das Sprite 2, die beide dann über den Bildschirm bewegt werden. Falls Sie nach Programmende mittels GRAPHIC 0 noch auf den Textbildschirm (40 Zeichen) umschalten, sehen Sie noch einen Unterschied zwischen Shapes und Sprites: Letztere bleiben auch hier aktiv.

Shapes aus Sprites bauen

Natürlich ist auch der umgekehrte Weg möglich: Nämlich Shapes aus Sprites zu konstruieren. Fügen Sie einfach mal an unser letztes Programm aus Listing 13 »SHAPE UND SPRITE« folgende Zeilen an:

```
130 CLR :REM LOESCHEN DER STRINGS
150 SPRSAV 1,A$:SPRSAV 2,B$
160 GSHAPE A$,150,100:GSHAPE B$,200,100
```

Über den Sinngehalt solch einer Zuordnung läßt sich natürlich streiten. Denkbar wäre es, das anzuwenden bei Speicherung eines Shape-Musters im Spritespeicher und anschließendem Zurücklesen dieser Information. Darüber hatten wir oben bei Aufbewahrung von Shapes für die Nachwelt schon kurz nachgedacht.

Wenn Shapes und Sprites zusammenstoßen

Kollisionen von Sprites mit Shapes werden vom Computer ebenso registriert, als wären es Zusammenstöße mit Textbestandteilen. Sowohl der COLLISION- als auch der BUMP Befehl können verwendet werden. In einer erweiterten Version unseres Programmes »SHAPE UND SPRITE« namens »SHAPE/SPR-KOLL« (Listing 14) werden alle Möglichkeiten, die wir in den letzten Abschnitten besprochen haben, ausprobiert. Dabei findet der COLLISION-Befehl Anwendung und Sprite 1 ist durch einen Joystick in Port 2 steuerbar. Bei jedem Zusammenstoß wechselt der Sprite die Farben.

Wann Shapes und wann Sprites?

Jeder von Ihnen kann eigentlich nun die Frage beantworten, wann welches der beiden Grafik-Objekte sinnvoll einzusetzen ist: Shapes und Sprites.

- Bewegte Objekte sollten Sprites sein
- Objekte, die in verschiedenen Grafikmodi (auch im

40-Zeichen-Textmodus) in gleicher Gestalt auftreten sollen, können nur Sprites sein.
- Statische Objekte in beliebiger Anzahl sind als Shapes gut zu verwirklichen
- Objekte, bei denen kein starr festgelegter Raster zugrundegelegt wird, können Shapes sein.

Nun sind Ihrer Phantasie keine Grenzen gesetzt: Bauen Sie sich ein Menü, in dem mit einem Sprite in Pfeil- oder Handform auf Symbole (Shapes) gedeutet wird, oder basteln Sie sich Schaltpläne aus vorgefertigten Einzelteilen (Shapes) zusammen, die Sie mit einem Kran-Sprite aussuchen und plazieren oder, oder. (H.Ponnath/sk)

```
10 REM ***** DATA-LADER FUER ALLE SPRITES *****      <A99>
15 FAST                                              <M2R>
20 GOSUB 100: BSAVE "SPRITES1",ON B0,P3584 TO P40
   95                                                <34G>
30 GOSUB 100: BSAVE "SPRITES2",ON B0,P3584 TO P40
   95                                                <14S>
40 GOSUB 100: BSAVE "SCENE1",ON B0,P3584 TO P4095    <0D2>
50 GOSUB 100: BSAVE "SCENE2",ON B0,P3584 TO P4095    <35E>
60 GOSUB 100: BSAVE "SCENE3",ON B0,P3584 TO P4095    <2TA>
65 SLOW                                              <UVO>
70 PRINT CHR$(147)"ALLE SPRITE-FILES SIND GESPEIC
   HERT"                                             <R32>
80 END                                               <PV6>
100 REM *** DATEN IN SPRITE-SPEICHER SCHREIBEN **
    *                                                <IIA>
110 FOR I=3584 TO 4095: READ A: POKE I,A: NEXT I     <IC9>
120 RETURN                                           <4S7>
63000 REM ******* SPRITE-DATEN *********             <D9K>
63010 REM *** DATAS VON FILE SPRITES1 ***            <D1L>
63020 DATA 000,000,000,000,000,000,000,170,000,00
      1,254,000,002,001,000,004,000,128,004,204,1
      28,004,000,128,060,048,240,036,000,144,036,
      132                                            <9IT>
63021 DATA 144,036,120,144,034,001,016,001,254,00
      0,000,132,000,000,132,000,000,132,000,000,1
      32,000,000,132,000,003,135,000,000,000,000,
      000                                            <U6G>
63022 DATA 000,000,000,000,000,000,000,170,000,00
      1,254,000,002,001,000,004,000,128,004,204,1
      28,004,000,128,060,048,224,036,000,160,036,
      132                                            <RGL>
63023 DATA 160,036,120,160,034,001,032,001,254,03
      2,000,132,032,000,130,000,000,129,000,000,1
      28,128,000,128,096,003,128,000,000,000,000,
      000                                            <NLK>
63024 DATA 000,000,000,000,000,000,000,170,000,00
      1,254,000,002,001,000,004,000,128,132,204,1
      28,132,000,128,252,048,224,004,000,160,004,
      132                                            <5GF>
63025 DATA 160,004,120,160,002,001,032,001,254,03
      4,000,132,034,000,067,254,000,064,000,000,0
      64,000,000,064,000,001,192,000,000,000,000,
      000                                            <U87>
63026 DATA 000,000,000,000,000,000,000,170,000,00
      1,254,000,002,001,008,004,000,136,004,204,1
      36,132,000,136,252,048,248,132,000,136,004,
      132                                            <6J3>
63027 DATA 136,004,120,142,002,001,000,001,254,00
      0,000,128,000,000,064,000,000,032,000,000,0
      32,000,000,032,000,000,224,000,000,000,000,
      000                                            <F2L>
63028 DATA 000,000,000,000,000,192,000,170,096,00
      1,254,048,002,001,024,004,000,140,132,204,1
      39,132,000,136,252,048,248,004,000,128,004,
      132                                            <45P>
63029 DATA 128,004,120,128,002,001,000,001,254,00
      0,000,128,000,000,064,000,000,032,000,000,0
      32,000,000,032,000,000,224,000,000,000,000,
      000                                            <PRE>
63030 DATA 000,000,000,000,000,000,255,170,000,16
      1,254,000,162,001,000,036,000,192,036,204,1
      60,036,000,144,060,048,248,004,000,128,004,
      132                                            <7HD>
63031 DATA 128,004,120,128,002,001,000,001,254,00
      0,000,128,000,000,064,000,000,032,000,000,0
      32,000,000,032,000,000,224,000,000,000,000,
      000                                            <JQ6>
63032 DATA 000,000,000,000,000,000,000,170,000,00
      1,254,000,002,001,000,004,000,192,004,204,1
      60,004,000,144,060,048,248,036,000,128,036,
      132                                            <Q0T>
63033 DATA 128,164,120,128,098,001,000,033,254,00
      0,016,128,000,008,064,000,004,032,000,002,0
      32,000,004,032,000,008,224,000,000,000,000,
      000                                            <86D>
63034 DATA 000,000,000,000,000,000,000,170,000,00
      1,254,000,002,001,000,004,000,128,004,204,1
      28,004,000,128,028,048,224,020,000,160,020,
      132                                            <N17>
63035 DATA 160,020,120,160,018,001,032,017,254,03
      2,016,132,032,031,252,032,000,039,224,000,0
      36,000,000,036,000,000,231,000,000,000,000,
      000                                            <9B8>
63039 REM *** DATAS VON FILE SPRITES2 ***            <CLJ>
63040 DATA 085,085,085,106,170,169,111,255,249,11
      1,255,249,111,255,249,109,085,121,109,170,1
      21,109,190,121,109,190,121,109,190,121,109,
      190                                            <5L8>
63041 DATA 121,109,190,121,109,190,121,109,190,12
      1,109,170,121,109,085,121,111,255,249,111,2
      55,249,111,255,249,106,170,169,085,085,085,
      000                                            <UCF>
63042 DATA 255,255,255,148,081,069,170,170,139,21
      3,021,085,162,138,041,213,085,081,168,170,1
      69,148,081,069,170,170,163,197,085,085,162,
      138                                            <LVR>
63043 DATA 041,149,085,021,138,170,171,148,081,06
      9,170,168,171,209,085,085,162,138,041,213,0
      69,085,170,042,171,148,017,069,255,255,255,
      000                                            <RRT>
63044 DATA 000,000,001,000,000,003,000,000,007,00
      0,000,013,000,000,025,000,000,049,000,000,0
      97,000,000,193,000,001,129,000,003,001,000,
      006                                            <QD9>
63045 DATA 001,000,012,001,000,024,129,000,048,12
      9,000,096,129,000,192,129,001,128,001,003,0
      00,001,006,000,001,012,000,001,031,255,255,
      000                                            <E8B>
63046 DATA 128,000,000,192,000,000,224,000,000,17
      6,000,000,152,000,000,140,000,000,134,000,0
      00,131,000,000,129,128,000,128,192,000,128,
      096                                            <K2I>
63047 DATA 000,148,048,000,148,024,000,148,012,00
      0,148,006,000,148,003,000,128,001,128,128,0
      00,192,128,000,096,128,000,048,255,255,248,
      000                                            <9CP>
63048 DATA 031,255,255,012,000,001,006,000,001,00
      3,002,161,001,130,161,000,194,161,000,098,1
      61,000,050,161,000,024,001,000,012,001,000,
      006                                            <S2U>
63049 DATA 001,000,003,001,000,001,129,000,000,19
      3,000,000,097,000,000,049,000,000,025,000,0
      00,013,000,000,007,000,000,003,000,000,001,
      000                                            <LKD>
63050 DATA 255,255,248,128,000,048,128,000,096,16
      8,128,192,168,129,128,165,003,000,165,006,0
      00,162,012,000,128,024,000,128,048,000,128,
      096                                            <LIT>
63051 DATA 000,128,192,000,129,128,000,131,000,00
      0,134,000,000,140,000,000,152,000,000,176,0
      00,000,224,000,000,192,000,000,128,000,000,
      000                                            <3H3>
63052 DATA 125,245,087,109,181,086,109,182,166,10
      9,182,246,109,182,246,109,182,246,109,182,2
      46,109,182,246,109,182,246,085,181,086,085,
      181                                            <U38>
63053 DATA 086,105,182,170,109,182,255,109,182,25
      5,109,182,255,109,182,255,109,182,255,109,1
      82,255,109,182,255,109,182,255,174,186,255,
      000                                            <SFH>
63054 DATA 095,255,214,103,255,102,105,253,166,11
      0,118,182,111,154,246,111,235,246,111,255,2
      46,111,255,246,111,219,250,111,219,255,111,
      085                                            <FVI>
63055 DATA 085,111,154,106,111,219,111,111,235,11
      1,111,255,111,111,255,111,111,255,111,111,2
      55,111,111,255,111,111,255,111,111,255,111,
      000                                            <DKD>
63059 REM *** DATAS VON FILE SCENE1 ***              <D5L>
63060 DATA 170,170,170,170,170,170,170,170,170,17
      0,170,170,165,085,090,165,085,090,165,085,0
      90,165,085,090,165,255,090,165,255,090,165,
      255                                            <1GL>
63061 DATA 090,165,255,090,165,255,090,165,255,09
      0,165,255,090,165,085,090,165,085,090,165,0
      85,090,165,085,090,170,170,170,170,170,170,
      000                                            <O7G>
63062 DATA 170,170,170,170,170,170,154,170,166,15
      4,170,166,165,085,090,165,085,090,167,085,2
      18,167,085,218,165,255,090,165,255,090,165,
      235                                            <CS1>
63063 DATA 090,165,235,090,165,235,090,165,255,09
      0,165,255,090,167,085,218,167,085,218,165,0
      85,090,165,085,090,154,170,166,154,170,166,
      000                                            <D9I>
63064 DATA 106,170,169,106,170,169,154,170,166,15
      4,170,166,173,085,122,173,085,122,167,085,2
      18,167,085,218,165,190,090,165,190,090,165,
      235                                            <667>
63065 DATA 090,165,235,090,165,235,090,165,190,09
      0,165,190,090,167,085,218,167,085,218,173,0
      85,122,173,085,122,154,170,166,090,170,165,
      000                                            <NOO>
63066 DATA 106,170,169,106,170,169,154,170,166,14
      9,085,086,173,085,122,173,085,122,167,085,2
      18,167,255,218,165,190,090,165,170,090,165,
      235                                            <4TB>
63067 DATA 090,165,195,090,165,235,090,165,170,09
      0,165,190,090,167,255,218,167,085,218,173,0
      85,122,173,085,122,149,085,086,090,170,165,
      000                                            <55K>
```

Listing 1. Alle »ALLE SPRITES«.
Dieser DATA-Lader erzeugt auf der Diskette die in den Beispielprogrammen benötigten Sprite-Files.
Bitte mit dem Checksummer 128 (Seite 158) eingeben.

```
63068 DATA 106,170,169,106,170,169,149,085,086,14
      9,085,086,157,085,118,157,085,118,151,255,2
      14,151,255,214,151,170,214,151,170,214,151,
      130                                              <S7U>
63069 DATA 214,151,130,214,151,130,214,151,170,21
      4,151,170,214,151,255,214,151,255,214,157,0
      85,118,157,085,118,149,085,086,090,170,165,
      000                                              <HOJ>
63070 DATA 106,170,169,106,170,169,181,085,094,18
      1,085,094,157,085,118,158,255,182,150,255,1
      50,151,170,214,151,040,214,151,040,214,151,
      130                                              <EHH>
63071 DATA 214,151,130,214,151,130,214,151,040,21
      4,151,040,214,151,170,214,150,255,150,158,2
      55,182,157,085,118,181,085,094,090,170,165,
      000                                              <3AL>
63072 DATA 106,170,169,085,085,085,181,085,094,18
      1,085,094,159,255,246,158,255,182,150,170,1
      50,151,170,214,151,040,214,151,000,214,151,
      130                                              <D7K>
63073 DATA 214,151,130,214,151,130,214,151,040,21
      4,151,040,214,151,170,214,150,255,150,158,2
      55,182,157,085,118,181,085,094,090,170,165,
      000                                              <1AT>
63074 DATA 085,085,085,085,085,085,117,085,093,12
      7,255,253,095,255,245,094,170,181,094,170,1
      81,094,000,181,094,000,181,094,020,181,094,
      020                                              <VR5>
63075 DATA 181,094,020,181,094,000,181,094,000,18
      1,094,170,181,094,170,181,095,255,245,127,2
      55,253,117,085,093,085,085,085,085,085,085,
      000                                              <D9H>
63079 REM *** DATAS VON FILE SCENE2 ***                <DLN>
63080 DATA 213,085,087,213,085,087,127,255,253,12
      3,255,237,123,255,237,124,170,061,124,170,0
      61,126,065,189,126,065,189,126,020,189,126,
      020                                              <JIJ>
63081 DATA 189,126,020,189,126,065,189,126,065,18
      9,124,170,061,124,170,061,123,255,237,123,2
      55,237,127,255,253,213,085,087,213,085,087,
      000                                              <463>
63082 DATA 213,085,087,239,255,251,111,255,249,12
      3,255,237,120,170,045,124,170,061,124,065,0
      61,126,065,189,126,085,189,126,020,189,126,
      020                                              <ANT>
63083 DATA 189,126,020,189,126,085,189,126,065,18
      9,124,065,061,124,170,061,120,170,045,123,2
      55,237,111,255,249,239,255,251,213,085,087,
      000                                              <T36>
63084 DATA 255,255,255,239,255,251,239,255,251,25
      0,170,175,250,170,175,248,000,047,248,000,0
      47,248,085,047,248,085,047,248,125,047,248,
      125                                              <AVI>
63085 DATA 047,248,125,047,248,085,047,248,085,04
      7,248,000,047,248,000,047,250,170,175,250,1
      70,175,239,255,251,239,255,251,255,255,255,
      000                                              <5AR>
63086 DATA 191,255,254,239,255,251,239,255,251,24
      2,170,143,242,170,143,249,000,111,249,000,1
      11,248,215,047,248,215,047,248,125,047,248,
      125                                              <EHA>
63087 DATA 047,248,125,047,248,215,047,248,215,04
      7,249,000,111,249,000,111,242,170,143,242,1
      70,143,239,255,251,239,255,251,191,255,254,
      000                                              <16N>
63088 DATA 191,255,254,239,255,251,234,170,171,24
      2,170,143,240,000,015,249,000,111,249,085,1
      11,248,215,047,248,255,047,248,125,047,248,
      125                                              <TPM>
63089 DATA 047,248,125,047,248,255,047,248,215,04
      7,249,085,111,249,000,111,240,000,015,242,1
      70,143,234,170,171,239,255,251,191,255,254,
      000                                              <IEB>
63090 DATA 191,255,254,234,170,171,234,170,171,22
      4,000,011,224,000,011,225,085,075,225,085,0
      75,225,255,075,225,255,075,225,255,075,225,
      255                                              <EB5>
63091 DATA 075,225,255,075,225,255,075,225,255,07
      5,225,085,075,225,085,075,224,000,011,224,0
      00,011,234,170,171,234,170,171,191,255,254,
      000                                              <BLM>
63092 DATA 191,255,254,202,170,163,202,170,163,22
      8,000,027,228,000,027,227,085,203,227,085,2
      03,225,255,075,225,255,075,225,235,075,225,
      235                                              <B0Q>
63093 DATA 075,225,235,075,225,255,075,225,255,07
      5,227,085,203,227,085,203,228,000,027,228,0
      00,027,202,170,163,202,170,163,191,255,254,
      000                                              <CE8>
63094 DATA 170,170,170,138,170,162,128,000,002,16
      4,000,026,165,085,090,163,085,202,163,255,2
      02,161,255,074,161,235,074,161,235,074,161,
      235                                              <ANU>
63095 DATA 074,161,235,074,161,235,074,161,255,07
      4,163,255,202,163,085,202,165,085,090,164,0
      00,026,128,000,002,138,170,162,170,170,170,
      000                                              <CP8>
63099 REM *** DATAS VON FILE SCENE3 ***                <E5P>
63100 DATA 170,170,170,128,000,002,128,000,002,13
      3,085,082,133,085,082,135,255,210,135,255,2
      10,135,170,210,135,170,210,135,170,210,135,
      170                                              <PRE>
63101 DATA 210,135,170,210,135,170,210,135,170,21
      0,135,255,210,135,255,210,133,085,082,133,0
      85,082,128,000,002,128,000,002,170,170,170,
      000                                              <3ID>
63102 DATA 128,000,002,133,085,082,135,255,210,13
      5,170,210,135,170,210,135,170,210,135,170,2
      10,135,170,210,135,170,210,135,170,210,135,
      170                                              <GVB>
63103 DATA 210,135,170,210,135,170,210,135,170,21
      0,135,170,210,135,170,210,135,170,210,135,1
      70,210,135,255,210,133,085,082,128,000,002,
      000                                              <ANJ>
63104 DATA 000,000,000,021,085,084,031,255,244,03
      0,170,180,030,170,180,030,170,180,030,170,1
      80,030,170,180,030,170,180,030,170,180,030,
      170                                              <2J1>
63105 DATA 180,030,170,180,030,170,180,030,170,18
      0,030,170,180,030,170,180,030,170,180,030,1
      70,180,031,255,244,021,085,084,000,000,000,
      000                                              <6VG>
63106 DATA 085,085,085,127,255,253,122,170,173,12
      2,170,173,122,170,173,122,170,173,122,170,1
      73,122,170,173,122,170,173,122,170,173,122,
      150                                              <FVC>
63107 DATA 173,122,170,173,122,170,173,122,170,17
      3,122,170,173,122,170,173,122,170,173,122,1
      70,173,122,170,173,127,255,253,085,085,085,
      000                                              <P64>
63108 DATA 085,085,085,111,255,249,122,170,173,12
      2,170,173,122,170,173,122,170,173,122,170,1
      73,122,170,173,122,150,173,122,150,173,122,
      085                                              <B2P>
63109 DATA 173,122,150,173,122,150,173,122,170,17
      3,122,170,173,122,170,173,122,170,173,122,1
      70,173,122,170,173,111,255,249,085,085,085,
      000                                              <UMD>
63110 DATA 169,085,106,175,255,250,191,170,254,19
      0,170,190,186,170,174,122,170,173,122,150,1
      73,122,150,173,122,150,173,122,085,173,121,
      125                                              <6N0>
63111 DATA 109,122,085,173,122,150,173,122,150,17
      3,122,150,173,122,170,173,186,170,174,190,1
      70,190,191,170,254,175,255,250,169,085,106,
      000                                              <V0D>
63112 DATA 170,170,170,175,255,250,191,255,254,19
      1,255,254,190,150,190,190,150,190,190,150,1
      90,190,150,190,189,085,126,189,125,126,189,
      255                                              <7DT>
63113 DATA 126,189,125,126,189,085,126,190,150,19
      0,190,150,190,190,150,190,190,150,190,191,2
      55,254,191,255,254,175,255,250,170,170,170,
      000                                              <GNP>
63114 DATA 170,170,170,175,255,250,191,255,254,19
      1,255,254,191,085,254,189,085,126,189,125,1
      26,189,255,126,189,195,126,189,195,126,189,
      195                                              <T5N>
63115 DATA 126,189,195,126,189,255,126,189,125,12
      6,189,085,126,189,085,126,191,085,254,191,2
      55,254,191,255,254,175,255,250,170,170,170,
      000                                              <CUH>
```

Listing 1. »Alle Sprites« (Schluß)

```
10 REM ***** TESTPROGRAMM 1 - COLLISION *****           <A99>
20 BLOAD "SPRITES2",ON B0                               <EBM>
30 SPRCOLOR 3,6                                         <6B0>
40 MOVSPR 7,300,100: MOVSPR 8,300,200                   <6BU>
50 SPRITE 7,1,2,0,0,0,1: SPRITE 8,1,1,1,1,1,1           <U60>
60 COLLISION 1,90                                       <L2R>
70 MOVSPR 8,+0,+5                                       <H1N>
80 GOTO 70                                              <PN9>
90 COLLISION 1                                          <6F4>
100 PRINT "KOLLISION ERFOLGT"                           <1D6>
110 COLLISION 1,130                                     <UL4>
120 RETURN                                              <4S7>
130 COLLISION 1                                         <E35>
140 PRINT "NOCH EINE KOLLISION"                         <G7Q>
150 COLLISION 1,130                                     <UT5>
160 RETURN                                              <4K7>
```

Listing 2. Kollision von Sprites.
Bitte mit dem Checksummer 128 (Seite 158) eingeben.

```
10 REM ***** TESTPROGRAMM 2 BUMP-BEFEHL *****             <A99>
20 BLOAD "SPRITES2",ON B0                                 <EBM>
30 SPRCOLOR 3,6                                           <6B0>
40 MOVSPR 7,100,100: MOVSPR 8,150,200                     <IAE>
50 SPRITE 7,1,2,0,0,0,1: SPRITE 8,1,1,1,1,1,1             <U60>
60 V=BUMP(1): IF V>0 THEN GOSUB 85                        <032>
70 MOVSPR 8,+1,+3                                         <GPG>
80 GOTO 60                                                <PV8>
85 PRINT : PRINT "BUMP(1) ERGIBT DEN WERT"V               <4FI>
90 FOR I=0 TO 7: A=2↑I: IF A AND V THEN PRINT "SP
   RITE "I+1"KOLLIDIERT                                   <NN7>
100 NEXT I                                                <RJ4>
110 RETURN                                                <KQ7>
```

Listing 3. Ein Programm zur BUMP-Abfrage
Bitte mit dem Checksummer 128 (Seite 158) eingeben.

```
10 REM ***** TESTPROGRAMM 3  JOY-BEFEHL *****          <A99>
20 BLOAD "SPRITES2",ON B0                              <EBM>
30 SPRCOLOR 3,6                                        <680>
40 MOVSPR 1,100,100: MOVSPR 8,200,200                  <UJU>
50 SPRITE 1,1,2,0,0,0,1: SPRITE 8,1,1,1,1,1,1          <UU3>
60 COLLISION 1,110                                     <BDV>
70 X=200: Y=200                                        <DIH>
80 ON JOY(2) GOSUB 190,200,210,220,230,240,250,26
   0                                                   <5GF>
90 MOVSPR 8,X,Y                                        <J91>
100 GOTO 80                                            <SR6>
110 COLLISION 1                                        <C35>
120 PRINT "KOLLISION ERFOLGT"                          <1T6>
130 COLLISION 1,150                                    <V12>
140 RETURN                                             <4G7>
150 COLLISION 1                                        <835>
160 PRINT "NOCH EINE KOLLISION"                        <I7Q>
170 COLLISION 1,150                                    <V93>
180 RETURN                                             <4B7>
190 X=X: Y=Y-1: RETURN                                 <V6G>
200 X=X+1: Y=Y-1: RETURN                               <731>
210 X=X+1: Y=Y: RETURN                                 <HUV>
220 X=X+1: Y=Y+1: RETURN                               <5R9>
230 X=X: Y=Y+1: RETURN                                 <SUS>
240 X=X-1: Y=Y+1: RETURN                               <JVH>
250 X=X-1: Y=Y: RETURN                                 <SUR>
260 X=X-1: Y=Y-1: RETURN                               <H7P>
```

Listing 4. Steuern mittels JOY

```
10 REM ***** VERGLEICH SPRITEDATEN *****               <A99>
20 REM IN SPRITESPEICHER UND IM STRING                 <AHA>
30 DIM A$(7)                                           <BKR>
40 BLOAD "SPRITES1",ON B0                              <EBM>
50 FOR J=0 TO 7                                        <4BK>
60 : PRINT CHR$(147) CHR$(18)J+1".SPRITE" CHR$(14
   6): PRINT : SPRSAV J+1,A$(J)                        <JBL>
70 : BANK 0: PRINT "DATENSPEICHER:"                    <OIG>
80 : FOR I=0 TO 63                                     <60G>
90 : : PRINT(PEEK(3584+64*J+I));                       <579>
100 : NEXT I: PRINT : PRINT "STRING:"                  <339>
110 : FOR K=1 TO LEN(A$(J))                            <ORA>
120 : : PRINT ASC(MID$(A$(J),K,1));                    <K92>
130 : NEXT K: PRINT : PRINT "BITTE TASTE DRUECKEN
   "                                                   <1Q1>
140 : GET KEY B$                                       <40E>
150 : PRINT                                            <JAU>
160 NEXT J                                             <BP5>
```

Listing 5. »VERGLEICH SPRDT«.
Programm zum Vergleichen von Spritemustern im Speicher und im String.

```
62999 REM ***.DATAS AUS SPRITESPEICHER ***             <BLF>
63000 FAST : BANK 0: FOR J=0 TO 7                      <BAM>
63001 A$(J)="": FOR I=0 TO 62: A$=STR$(PEEK(3584+
      64*J+I))                                         <UJU>
63002 A$=RIGHT$(A$,LEN(A$)-1): A$="000"+A$: A$=RI
      GHT$(A$,3)                                       <V5Q>
63003 A$(J)=A$(J)+A$+",": NEXT I                       <2AP>
63004 A$(J)=A$(J)+"000"                                <V64>
63005 NEXT J: SLOW                                     <QM4>
63006 FOR J=0 TO 7: PRINT CHR$(147);                   <ARM>
63007 PRINT 63020+2*J"DATA" LEFT$(A$(J),127)           <U3H>
63008 PRINT 63021+2*J"DATA" RIGHT$(A$(J),127): PR
      INT "GOTO63009": STOP                            <GGI>
63009 NEXT J: PRINT CHR$(147);                         <75L>
63010 PRINT "63040 FORI=3584TO4095:READA:POKEI,A:
      NEXTI:RETURN"                                    <K0K>
63011 PRINT "DELETE 63000-63011": STOP                 <KBP>
```

Listing 6. »SPRITEDATAS«.
Erzeugen von DATA-Zeilen aus Spritedaten

```
10 REM ***** VIER SPRITES *****                        <A99>
20 GOTO 110                                            <NE7>
30 Y=Y-3: RETURN                                       <Q04>
40 X=X+3: Y=Y-3: RETURN                                <FB1>
50 X=X+3: RETURN                                       <3G4>
60 X=X+3: Y=Y+3: RETURN                                <DJ9>
70 Y=Y+3: RETURN                                       <A04>
80 X=X-3: Y=Y+3: RETURN                                <JMH>
90 X=X-3: RETURN                                       <JC4>
100 X=X-3: Y=Y-3: RETURN                               <P3P>
110 BLOAD "SPRITES2",ON B0                             <BR6>
120 PRINT CHR$(147): COLOR 0,1: COLOR 4,1: COLOR
   5,8                                                 <PPD>
130 X=50: Y=60                                         <D40>
140 MOVSPR 3,X,Y: MOVSPR 4,X+125,Y: MOVSPR 5,X,Y+
   122: MOVSPR 6,X+125,Y+122                           <2CO>
150 SPRITE 3,1,8,1: SPRITE 4,1,8,1: SPRITE 5,1,8,
   1: SPRITE 6,1,8,1                                   <I3I>
160 DO UNTIL A=101                                     <TEJ>
```

Listing 7. »VIER SPRITES«.
Demonstration zum Koppeln von Sprites.

```
170 : MOVSPR 4,-1,Y: MOVSPR 5,X,-1: MOVSPR 6,-1,-
   1                                                   <6LN>
180 : A=A+1                                            <V7D>
190 LOOP                                               <V7P>
200 BANK 15: SYS 65520,10,20: A$="JOYSTICK IN POR
   T2"                                                 <HDM>
210 DO UNTIL JOY(2)=128                                <LHM>
220 : COLOR 5,3                                        <44F>
230 : GOSUB 330                                        <O3T>
240 : COLOR 5,1                                        <7CE>
250 : GOSUB 330                                        <R3T>
260 LOOP : X=RSPPOS(3,0): Y=RSPPOS(3,1)                <87U>
270 COLOR 5,8: PRINT CHR$(147): FOR I=1 TO 50: NE
   XT                                                  <9BQ>
280 DO                                                 <SNJ>
290 : ON JOY(2) GOSUB 30,40,50,60,70,80,90,100         <4TA>
300 : MOVSPR 3,X,Y: MOVSPR 4,X+24,Y: MOVSPR 5,X,Y
   +21: MOVSPR 6,X+24,Y+21                             <RAL>
310 LOOP UNTIL JOY(2)=128                              <8O1>
320 END                                                <146>
330 SYS 65520,10,20: PRINT A$: FOR K=1 TO 50: NEX
   T K: RETURN                                         <PGH>
```

Listing 7. »VIER SPRITES« (Schluß)

```
10 REM ***** EIN SPRITE-TRICK *****                    <A99>
20 BLOAD "SPRITES1",ON B0                              <M8M>
30 PRINT CHR$(147)                                     <HA3>
40 FOR I=1 TO 8: MOVSPR I,200,100: NEXT I              <VR0>
50 FOR I=1 TO 10: FOR J=1 TO 8                         <8UP>
60 IF J=1 THEN SPRITE 8,0: ELSE SPRITE J-1,0           <BE4>
70 SPRITE J,1,I+1,1,1,1,0                              <EVA>
80 FOR K=1 TO 25: NEXT K: NEXT J                       <VGO>
90 NEXT I                                              <004>
100 SPRITE 8,0: SPRITE 1,0                             <A50>
```

Listing 8. »SPRITE-TRICK«. Ein kleiner Trickfilm.

```
10 REM ***** 24 SPRITES *****                          <A99>
20 I=0: J=1: K=2: L=3: M=4: N=5: O=6: P=7: B$="":
   DIM A$(24)                                          <E5G>
30 BLOAD "SCENE1"                                      <L7C>
40 FOR I=1 TO 8: SPRSAV I,A$(I): NEXT I                <7PU>
50 BLOAD "SCENE2"                                      <5RF>
60 FOR I=1 TO 8: SPRSAV I,A$(I+8): NEXT I              <GEF>
70 BLOAD "SCENE3"                                      <LJE>
80 FOR I=1 TO 8: SPRSAV I,A$(I+16): NEXT I             <PJT>
90 PRINT CHR$(147)                                     <HU1>
100 SPRCOLOR 3,6: COLOR 0,1: COLOR 4,1: COLOR 5,8      <6OJ>
110 MOVSPR 1,62,86: MOVSPR 2,111,86: MOVSPR 3,160
   ,86: MOVSPR 4,209,86: MOVSPR 5,62,129               <MH7>
120 MOVSPR 6,111,129: MOVSPR 7,160,129: MOVSPR 8,
   209,129                                             <QP6>
130 SPRITE 1,1,8,1,1,1,1: SPRITE 2,1,14,1,1,1,1:
   SPRITE 3,1,2,1,1,1,1                                <2BT>
140 SPRITE 4,1,11,1,1,1,1: SPRITE 5,1,13,1,1,1,1:
   SPRITE 6,1,5,1,1,1,1                                <KVA>
150 SPRITE 7,1,1,1,1,1,1: SPRITE 8,1,16,1,1,1,1        <VEJ>
160 I=0                                                <LEN>
170 DO                                                 <LMN>
180 : I=I+1: IF I>24 THEN I=1: SPRSAV A$(I),1: EL
   SE SPRSAV A$(I),1                                   <H04>
190 : J=J+1: IF J>24 THEN J=1: SPRSAV A$(J),2: EL
   SE SPRSAV A$(J),2                                   <Q1E>
200 : K=K+1: IF K>24 THEN K=1: SPRSAV A$(K),3: EL
   SE SPRSAV A$(K),3                                   <368>
210 : L=L+1: IF L>24 THEN L=1: SPRSAV A$(L),4: EL
   SE SPRSAV A$(L),4                                   <S5Q>
220 : M=M+1: IF M>24 THEN M=1: SPRSAV A$(M),5: EL
   SE SPRSAV A$(M),5                                   <L4S>
230 : N=N+1: IF N>24 THEN N=1: SPRSAV A$(N),6: EL
   SE SPRSAV A$(N),6                                   <U5M>
240 : O=O+1: IF O>24 THEN O=1: SPRSAV A$(O),7: EL
   SE SPRSAV A$(O),7                                   <76G>
250 : P=P+1: IF P>24 THEN P=1: SPRSAV A$(P),8: EL
   SE SPRSAV A$(P),8                                   <AAQ>
260 : GET B$: IF B$="←" THEN EXIT                      <J8D>
270 LOOP                                               <V3P>
280 END                                                <H66>
```

Listing 9. »24 SPRITES«

```
10 REM ***** SSHAPE TABELLE *****                      <A99>
20 FAST : OPEN 1,4: PRINT#1,"Y-DIFF.","X-DIFF.","
   STRINGLAENGE"                                       <HU0>
30 DEF FN H(X)=INT((ABS(X)+1)/8+.99)*(ABS(Y)+1)+4      <FLF>
40 Y=0                                                 <TNN>
50 DO UNTIL Y=200                                      <9CJ>
60 : Y=Y+1: X=0                                        <315>
70 : DO UNTIL X=320                                    <IOT>
80 : : X=X+1                                           <GEN>
90 : : Z=FN H(X): IF Z>255 THEN PRINT#1,Y1,X1,Z1:
   EXIT                                                <0DC>
100 : : X1=X: Y1=Y: Z1=Z                               <QFT>
110 : LOOP                                             <S3P>
120 LOOP                                               <J29>
130 CLOSE 1: SLOW : END                                <15D>
```

Listing 10. »SSHAPE-TABELLE«. Eine Hilfe für Shapes: Eine Tabelle wird gedruckt

```
10 REM *** DIE GSHAPE-MODI ***                      <A99>
20 REM ERSTELLEN EINES SHAPE                        <AHA>
30 COLOR 0,1: COLOR 1,6: COLOR 4,1: GRAPHIC 1,1     <OPE>
40 CIRCLE 1,10,10,10,10: BOX 1,0,0,20,20: PAINT 1
   ,1,1: PAINT 1,19,1: PAINT 1,1,19                 <191>
50 PAINT 1,19,19: CIRCLE 1,10,10,7,7: DRAW 1,10,0
   TO 10,20: DRAW 1,0,10 TO 20,10                   <EGH>
60 SSHAPE A$,0,0,20,20                              <GLS>
70 SCNCLR 1                                         <P7U>
80 REM BILDSCHIRMHINTERGRUND                        <B10>
90 DX=5: DY=5: X=0: Y=0                             <VUT>
100 DO WHILE X<511                                  <9J6>
110 : X=X+DX: Y=Y+DY                                <23H>
120 : DRAW 1,0,Y TO X,0                             <FHC>
130 LOOP                                            <N2P>
140 REM ABBILDEN DES SHAPE                          <JIE>
150 FOR I=0 TO 4                                    <P25>
160 : GSHAPE A$,10+55*I,100,I                       <E5F>
170 NEXT I                                          <3K4>
180 REM KOMMENTAR                                   <GI2>
190 COLOR 1,3                                       <MG7>
200 FOR I=0 TO 4                                    <OI5>
210 : B$=RIGHT$(STR$(I),1)                          <2D9>
220 : CHAR 1,2+7*I,18,B$                            <K1T>
230 NEXT I                                          <JM4>
240 CHAR 1,1,4,"DIE GSHAPE-MODI UND IHRE WIRKUNGE
    N:",1                                           <0QO>
250 CHAR 1,1,23,"BITTE EINE TASTE"                  <IUA>
260 GET KEY A$                                      <M4D>
270 GRAPHIC 0,1                                     <3IH>
280 END                                             <H66>
```

Listing 11. »GSHAPE-MODI«. Die 5 Shape-Darstellungsformen.

```
10 REM ***** SHAPE ERSTELLEN *****                  <A99>
20 COLOR 0,1: COLOR 4,1: COLOR 1,6: GRAPHIC 1,1:
   B$=CHR$(13)                                      <5PS>
30 CIRCLE 1,10,10,10,10: SSHAPE A$,0,0,20,20: GOS
   UB 230                                           <5Q3>
40 SCRATCH "SHAPE1"                                 <5SP>
50 REM ***** SHAPES AUF DISKETTE SPEICHERN *****    <B9D>
60 DOPEN #1,"SHAPE1",W: PRINT#1                     <NKH>
70 FOR I=1 TO LEN(A$): PRINT#1,ASC(MID$(A$,I,1))B
   $: NEXT I: PRINT#1,"ENDE"B$                      <B99>
80 DCLOSE #1                                        <6T5>
90 REM ***** VARIABLE LOESCHEN UND NEUSTART *****   <B91>
100 CLR : GRAPHIC 1,1: SLEEP 2: RUN 120             <R69>
110 REM ***** SHAPES WIEDER VON DISKETTE HOLEN **
    ***                                             <IQB>
120 DOPEN #1,"SHAPE1"                               <G2K>
130 DO                                              <HM7>
140 : INPUT#1,B$                                    <HHM>
150 : : IF B$="ENDE" THEN EXIT                      <6M5>
160 : B=VAL(B$): B$=CHR$(B): A$=A$+B$               <L0V>
170 LOOP                                            <70P>
180 DCLOSE #1: GOSUB 230                            <3Q7>
190 REM ***** SHAPE ZEICHNEN *****                  <GQ3>
200 GSHAPE A$,100,100                               <D4A>
210 END                                             <34E>
220 REM ***** KONTROLLUNTERPROGRAMM *****           <JIE>
230 FOR I=1 TO LEN(A$): PRINT ASC(MID$(A$,I,1));:
     NEXT I: PRINT : PRINT LEN(A$): RETURN          <T7E>
```

Listing 12. »SHAPE SPEICHERN«. Shapes retten auf Diskette oder Kassette.

```
1 REM ****** ERZEUGUNG VON DATA-ZEILEN AUS SHAPE-
  STRINGS (HIER A$) ******                          <640>
2 REM                                               <6BP>
3 REM ****** PROBESHAPE ERZEUGEN ******             <6CP>
4 REM                                               <6GQ>
5 COLOR 0,1: COLOR 4,1: COLOR 1,6: GRAPHIC 1,1: C
  IRCLE 1,10,10,10,10: SSHAPE A$,0,0,20,20          <E58>
6 STOP                                              <CPJ>
7 REM ****** ERZEUGEN DER DATAZEILEN ******         <6SR>
8 GOTO 63000                                        <0O4>
9 REM ****** NEUES PROGRAMM ZUR KONTROLLE DER DAT
  AS ******                                         <74S>
10 COLOR 0,1: COLOR 4,1: COLOR 1,3: GRAPHIC 1,1:
```

Listing 13. »SHAPE DATA«. Shapes in DATA-Zeilen ablegen.

```
   GOSUB 63015                                      <PQO>
15 GSHAPE C$,100,100                                <KBD>
20 END                                              <DTM>
62999 REM ****** ES FOLGT DER DATAGENERATOR BZW D
      IE DATAS ******                               <BLF>
63000 PRINT CHR$(147)63020"DATA" LEN(A$)            <BG7>
63001 I=1: K=0                                      <SN3>
63002 DO                                            <CPN>
63003 : Z=0: PRINT 63021+K"DATA";                   <IKH>
63004 : DO WHILE I<LEN(A$)                          <KMT>
63005 : : PRINT ASC(MID$(A$,I,1))",";: I=I+1: Z=Z
      +1                                            <9AA>
63006 : : IF Z=10 THEN EXIT                         <7UU>
63007 : LOOP                                        <JV9>
63008 : PRINT ASC(MID$(A$,I,1)): I=I+1: K=K+1       <M6J>
63009 LOOP WHILE I<LEN(A$)                          <BC4>
63010 PRINT 63021+K"DATA" ASC(MID$(A$,I,1))         <PQ7>
63011 PRINT "63015 READA:FORI=1TOA:READB:B$=CHR$(
      B):C$=C$+B$:NEXTI:RETURN"                     <759>
63012 PRINT "DELETE63000-63012"                     <RKM>
```

Listing 13. »SHAPE DATA« (Schluß)

```
10 REM ***** SHAPES ZU SPRITES UMWANDELN *****      <A99>
15 REM ********** MULTICOLOR-SPRITES **********     <ATB>
20 COLOR 0,1: COLOR 1,6: COLOR 2,2: COLOR 3,3: CO
   LOR 4,1: COLOR 5,2                               <Q6Q>
30 GRAPHIC 3,1: CIRCLE 1,5,10,5,10: BOX 2,0,0,10,
   20: DRAW 3,0,10 TO 10,10                         <96G>
40 DRAW 3,5,0 TO 5,20                               <A46>
50 SSHAPE A$,0,0,11,20                              <KLC>
60 SPRSAV A$,1: SPRCOLOR 6,3: MOVSPR 1,100,100: S
   PRITE 1,1,2,0,0,0,1                              <BK0>
65 SLEEP 2                                          <AL7>
70 REM ********** NORMALER HIRESSPRITE **********   <BPF>
80 GRAPHIC 1,1: CIRCLE 1,50,10,10,10: CIRCLE 1,50
   ,10,7,7: CIRCLE 1,50,10,4,4                      <DEB>
90 SSHAPE B$,40,0,63,20: GSHAPE A$,0,0              <SSL>
100 SPRSAV B$,2: MOVSPR 2,200,200: SPRITE 2,1,6     <PQ5>
110 MOVSPR 1,0#4: MOVSPR 2,90#8                     <HDB>
```

Listing 14. »SHAPE UND SPRITE«. Shapes werden zu Sprites.

```
10 REM ***** KOLLISION VON SPRITES UND SHAPES ***
   **                                               <A99>
20 REM ***** SHAPES ZU SPRITES UMWANDELN *****      <AHA>
30 REM ********** MULTICOLOR-SPRITES **********     <APB>
40 COLOR 0,1: COLOR 1,6: COLOR 2,2: COLOR 3,3: CO
   LOR 4,1: COLOR 5,2                               <A0Q>
50 GRAPHIC 3,1: CIRCLE 1,5,10,5,10: BOX 2,0,0,10,
   20: DRAW 3,0,10 TO 10,10                         <P2G>
60 DRAW 3,5,0 TO 5,20                               <AC7>
70 SSHAPE A$,0,0,11,20                              <ML4>
80 SPRSAV A$,1: SPRCOLOR 6,3: MOVSPR 1,100,100: S
   PRITE 1,1,2,0,0,0,1                              <5K0>
90 SLEEP 2                                          <IE7>
100 REM ********** NORMALER HIRESSPRITE *********
    *                                               <IIA>
110 GRAPHIC 1,1: CIRCLE 1,50,10,10,10: CIRCLE 1,5
    0,10,7,7: CIRCLE 1,50,10,4,4                    <637>
120 SSHAPE B$,40,0,63,20: GSHAPE A$,0,0             <6NS>
130 SPRSAV B$,2: MOVSPR 2,200,200: SPRITE 2,1,6     <9S5>
140 MOVSPR 1,0#4: MOVSPR 2,90#8                     <G5B>
150 SLEEP 2                                         <KI7>
160 REM ********** LOESCHEN DER STRINGS *********
    *                                               <J2C>
170 CLR                                             <2CB>
180 REM ********** UEBERTRAGEN VON SPRITES IN SHA
    PES **********                                  <GI2>
190 SPRSAV 1,A$: SPRSAV 2,B$                        <67S>
200 GSHAPE A$,150,100: GSHAPE B$,200,100            <JOE>
210 REM ********** STEUERN VON SPRITE 1 *********
    *                                               <JAD>
220 CHAR 1,20,0,"JOYSTICK PORT 2"                   <466>
230 REM ********** KOLLISION M. SHAPES **********   <JQF>
240 MOVSPR 1,0#0                                    <981>
250 X=RSPPOS(1,0): Y=RSPPOS(1,1): I=6: J=3          <CSM>
260 COLLISION 2,370                                 <BLS>
270 ON JOY(2) GOSUB 290,300,310,320,330,340,350,3
    60                                              <D49>
280 GOTO 270                                        <QM7>
290 Y=Y-2: MOVSPR 1,X,Y: RETURN                     <D2H>
300 X=X+1: Y=Y-1: MOVSPR 1,X,Y: RETURN              <6EB>
310 X=X+2: MOVSPR 1,X,Y: RETURN                     <NHH>
320 X=X+1: Y=Y+1: MOVSPR 1,X,Y: RETURN              <ABR>
330 Y=Y+2: MOVSPR 1,X,Y: RETURN                     <331>
340 X=X-1: Y=Y+1: MOVSPR 1,X,Y: RETURN              <97Q>
350 X=X-2: MOVSPR 1,X,Y: RETURN                     <QG1>
360 X=X-1: Y=Y-1: MOVSPR 1,X,Y: RETURN              <52A>
370 COLLISION 2                                     <EF4>
380 I=I+1: IF I>16 THEN I=1                         <DKG>
390 J=J+3: IF J>16 THEN J=1                         <UBP>
400 SPRCOLOR I,J                                    <269>
410 COLLISION 2,370                                 <O1U>
420 RETURN                                          <487>
```

Listing 15. »SHAPE/SPR-KOLL«. Zusammenspiel von Sprites und Shapes.

Lassen Sie Ihren C128 Landschaften und Figuren als dreidimensionale Höhenreliefkarten zeichnen und staunen Sie, wie Berge und Täler auf Bildschirm und Papier entstehen.

Das Programm »3-D-Landkarten« (Listing 1) erzeugt nach Eingabe von Koordinatenpunkten die planare Projektion eines Landschaftsreliefs. Neben der Bildschirmdarstellung ist auch eine Hardcopy-Routine für Ausdrucke in einfacher oder doppelter Größe vorgesehen; die erzeugten Grafiken können gespeichert und beliebig weiterverwendet werden. Die Bedienung des Programms ist dank einer guten Dialogführung sehr einfach, allerdings sollte man bei komplexeren Bildern etwas Zeit für Dateneingabe und Berechnungen einkalkulieren.

Bevor Sie mit »3-D-Landkarten« arbeiten können, benötigen Sie eine topographische Landkarte, deren Höhenlinien in Form einzelner Koordinatenpaare eingegeben werden.

COMPUTER LANDSCHA DRITTER DIMENSI

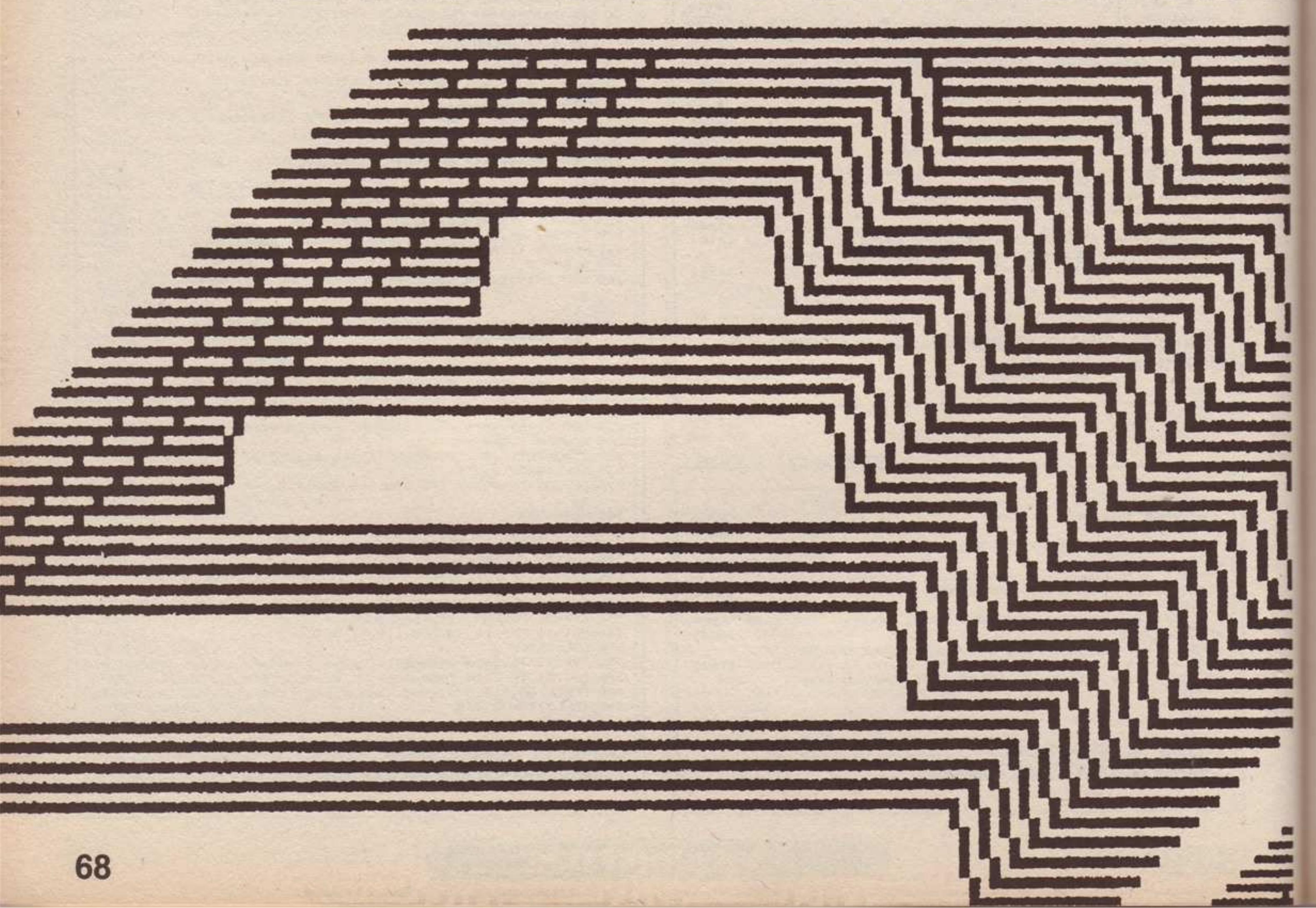




Bild 1. 3D-Darstellung einer Pyramide in Hires-Auflösung.

Zur Bestimmung dieser Punkte legen Sie auf die Landkarte ein transparentes Papier mit Millimeterraster auf. Noch günstiger ist es, die Vorlage mit Hilfe eines Fotokopiergerätes auf Millimeterpapier zu übertragen. Wählen Sie nun den gewünschten Bereich aus der Karte aus und legen Sie fest, welche Werte verwendet werden sollen.

Nach dem Programmstart fordert Sie »3-D-Landkarten« zunächst auf, das Koordinatensystem einzurichten, also den Wertebereich der X- und Y-Koordinaten festzulegen. Dabei werden Minimal- und Maximalwerte vorgegeben. Anschließend geben Sie Punkt für Punkt die Koordinatenwerte der einzelnen Höhenlinien Ihrer Karte ein. Wählen Sie den Abstand zwischen zwei Einzelpunkten nicht zu gering. Andernfalls kostet die Übertragung nicht nur Zeit, sondern überfordert vielleicht auch die Auflösungsmöglichkeiten Ihres Bildschirms. So ergibt sich beispielsweise bei einer Bildschirmskalierung von »1000/1000« für einen Abstand zweier Punkte mit dem Wert »5« lediglich ein Pixel in der Darstellung.

Alle Höhenlinien müssen in sich geschlossen sein, das heißt erster und letzter Punkt müssen jeweils dieselben Koordinatenwerte aufweisen. Andernfalls kann das Programm keine Abgrenzungen erkennen. Bei den Berechnungen wird davon ausgegangen, daß der Punkt mit dem Wertepaar »0/0« zur tiefsten Fläche der Abbildung gehört.

Sollte dies einmal nicht der Fall sein, so sind die Zeilen 300 und 1255 in Listing 1 entsprechend zu ändern.

Wenn Sie die Eingabe einer Höhenlinie abgeschlossen haben und weitere Linien derselben Höhe eingeben möchten, so beantworten Sie die Frage nach der X-Koordinate mit <P> (Linie an einem anderen Punkt fortsetzen, Zeile 1000). Wollen Sie mit der nächsthöheren Linie fortfahren,

Eingabe der Höhenlinien

wählen Sie hier <N>. Haben Sie die Eingabe aller Linien beendet, so geben Sie <Q> ein. Sollte Ihnen bei der Eingabe der Werte ein Fehler unterlaufen, so läßt sich dieser leicht beheben. Geben Sie einfach als nächstes Wertepaar noch einmal die richtigen Koordinaten ein und fahren Sie dann fort. Die dadurch entstehende Einzellinie wird später bei der Umrechnung eliminiert.

Nach dieser Eingabe der Koordinaten muß die Überhöhung, also die Streckung des Profils in vertikaler Richtung, festgelegt werden (Zeile 350 ff.). Der Zahlenwert hier bezieht sich auf den Abstand zwischen zwei Höhenstufen der dreidimensionalen Grafik in Pixeleinheiten. Anschließend lädt das Programm sukzessive die vorher angelegten Höhenkarten nach und berechnet die tatsächlichen Vertikalwerte für das Datenfeld »p%« (Zeile 380 ff.). Aus diesen Daten wird sodann (ab Zeile 600 bzw. 700) die Grafik erstellt. Das fertige Bild kann unter einem beliebigen Namen (maximal 13 Zeichen!) gespeichert werden (Zeile 800 f.), die dazugehörige Farbdatei erhält automatisch die Bezeichnung »Name/c«. Mit den Befehlen

`BLOAD "Name", ON B0`

und

`BLOAD "Name/c", ON B0`

holen Sie diese Files wieder in den Speicher und bringen Sie mit GRAPHIC 3 und GRAPHIC 1 wieder auf den Schirm.

Neben Landschaftsprofilen, deren Eingabe viel Zeit beanspruchen kann, eignet sich das Programm ebensogut zur Darstellung der Projektionen geometrischer Körper. Dabei ist die Anzahl der einzugebenden Wertepaare erheblich geringer. Als Beispiel für die Eingabe wird hier eine einfache Pyramide (Bild 1) vorgestellt, die sich für eine Skalierung des Bildschirms in den Bereichen »1000/1000« wie folgt bestimmen läßt:

1. Linie: 100/100 - 900/100 - 900/900 - 100/900 - 100/100
2. Linie: 200/200 - 800/200 - 800/800 - 200/800 - 200/200
3. Linie: 300/300 - 700/300 - 700/700 - 300/700 - 300/300
4. Linie: 400/400 - 600/400 - 600/600 - 400/600 - 400/400

(Ralf Kopp/Dr. Rudolf Egg/sk)

```
1 REM "***********************************"                 <64D>
2 REM "*{33SPACE}*"                                           <68P>
3 REM "*{3SPACE}HOEHENRELIEF-LANDKARTEN{7SPACE}*"             <6CP>
4 REM "*{33SPACE}*"                                           <6GQ>
5 REM "* FUER C128 UND COMMODORE-DRUCKER *"                   <6KQ>
6 REM "*{33SPACE}*"                                           <6OR>
7 REM "*{3SPACE}02/88{8SPACE}R.KOPP{11SPACE}*"                <6SR>
8 REM "*{33SPACE}*"                                           <70S>
9 REM "***********************************"                 <74S>
10 IF PEEK(215)=0 THEN 100                       : REM P
   RUEFUNG 40ER/80ER-SCHIRM                                   <N0B>
15 PRINT "{CLR,5DOWN,RVSON,4SPACE}A C H T U N G !
   {3SPACE}"                                                  <E5H>
20 PRINT "{3DOWN,RVSON} SCHALTE UM AUF 40-ER SCHI
   RM{2SPACE}"                                                <1UH>
25 SLEEP 5                                                    <A14>
30 SCNCLR 5                                                   <P7S>
35 PRINT CHR$(27)+"X"                            : REM UM
   SCHALTEN AUF 40 ZEICHEN                                    <23D>
100 GRAPHIC 0,1                                               <2IH>
105 PRINT CHR$(14)                                            <6T8>
110 PRINT "{CLR,4DOWN,5RIGHT,RVSON,2SPACE}R E L{S
    HF.SPACE}I{SHF.SPACE}E{SHF.SPACE}F - K A R T
    E N{3SPACE}"                                              <J8L>
115 PRINT "{3DOWN}SIE GEBEN DIE HOEHENLINIEN EINE
    R LAND-"                                                  <SF7>
120 PRINT "KARTE IN EINZELKOORDINATEN EIN."                   <EK9>
125 PRINT "BEGINNEN SIE MIT DER TIEFSTEN UND"                 <20A>
130 PRINT "FAHREN SIE AUFSTEIGEND FORT."                      <9KR>
135 PRINT "ALLE LINIEN MUESSEN IN SICH GESCHLOSSE
    N" : REM ABGRENZUNG IST ERFORDER-                         <2G8>
140 PRINT "SEIN, GGF. DURCH RANDPARALLELEN."
      : REM LICH,DA DIE HOEHENSTUFEN                          <3E4>
145 :
      : REM AUS DEM BILDSCHIRM                                <50L>
150 :
      : REM ERRECHNET WERDEN                                  <6GR>
155 PRINT "KOORDINATEN MIT 0 SIND VERBOTEN. DER "
      : REM 0/0 WIRD ALS AUSGANGS-                            <270>
160 PRINT "PUNKT 0/0 GEHOERT ZUR TIEFSTEN FLAECHE
    ." : REM PUNKT FUER PAINT BENUTZT                         <JIJ>
165 PRINT "SIE KOENNEN JETZT IHR KOORDINATENSYSTE
    M"                                                        <15E>
170 PRINT "EINRICHTEN."                                       <IF4>
175 PRINT "X-BEREICH VON 1 BIS ?"                : REM S
    KALIERUNG                                                 <B0A>
180 INPUT "(MIND.161/MAX.1023)";XM                            <S57>
185 PRINT "Y-BEREICH VON 1 BIS ?"                             <U2R>
190 INPUT "(MIND.201/MAX.1023)";YM                            <KD6>
195 S=1                                          : REM ZA
    EHLER(S) DER AKTUELLEN LINIE                              <1D2>
200 GRAPHIC 4,1                                               <6IP>
205 COLOR 0,1: COLOR 1,9: COLOR 2,8: COLOR 3,4: C
    OLOR 4,1 : REM FARBEN FUER VERTIKALE                      <ES8>
210 :
    : REM UND HORIZONTALE FLAECHEN                            <4GJ>
215 PRINT : PRINT "WIR BEGINNEN MIT DER EINGABE"              <SCT>
220 PRINT "DER ";S;". HOEHENLINIE"                            <E30>
225 SLEEP 5                                                   <C94>
230 SCALE 1,2*XM,YM                                           <0P6>
235 GOSUB 1000                                                <IJC>
240 GOSUB 1100                                                <GRE>
245 GOTO 1200                                                 <HBG>
300 LOCATE 0,0                                                <TMG>
305 GRAPHIC 3                                                 <A70>
310 PAINT 2,1,1                                               <P5N>
315 S$=STR$(S): N$="HOEHE"+S$                                 <5SB>
320 BSAVE(N$),ON B0,P08192 TO P16383                          <QFP>
325 GRAPHIC 0,1                                               <62H>
330 PRINT "{5DOWN}DIE FOLGENDE BERECHNUNG D. HOEH
    ENPROFILS"                                                <R44>
335 PRINT "{DOWN}ERFORDERT LANGE RECHENZEITEN."               <JB0>
340 PRINT "{DOWN}WUENSCHEN SIE HIERFUER DEN FAST-
    MODUS{3SPACE}(J/N)";                                      <0D0>
345 INPUT F$                                                  <0L8>
350 PRINT "{DOWN}WELCHE UEBERHOEHUNG DES PROFILS
    "; : REM UEBERHOEHUNG IST DER ABSTAND                     <6B5>
355 INPUT U                                         : R
    EM ZW.2.STUFEN.1=1 PIXEL                                  <7HK>
360 IF U*(S+1)>50 THEN PRINT "GRAFIK WIRD ZU HOCH
    .BITTE NEU WAEHLEN!": GOTO 350                            <00F>
365 PRINT "{3DOWN,RVSON}BERECHNUNG{SHF.SPACE}BEGI
    NNT !"                                                    <COV>
370 SLEEP 2                                                   <KU7>
375 IF F$="J" THEN FAST                                       <9ON>
380 GRAPHIC 3,1: K=1                                          <615>
385 DIM P%(100,80)                                            <3AF>
390 K$=STR$(K): N$="HOEHE"+K$                                 <18L>
395 BLOAD(N$),ON B0                        : REM NACHLADE
    N DER VERSCH. HOEHENKARTEN                                <4L4>
400 FOR A=8192 TO 15872 STEP 320                              <RVL>
405 FOR B=0 TO 7 STEP 2                                       <GCC>
410 FOR C=0 TO 159 STEP 4                                     <4P6>
415 D=A-8192                                                  <ANQ>
420 DA=D/40                                         : REM AU
    F 1/4 VERKLEINERN UND                                     <SP9>
425 DY=(DA+B)/2                                               <QT6>
430 DX=C/2                                          : REM UM
    RECHNEN AUF ABSOLUTE                                      <E8L>
435 BP=A+B+8*(INT(C/4))                                       <N57>
440 H=PEEK(BP)                                      : REM HO
    EHENWERTE                                                 <DGE>
445 IF H>127 AND P%(DY,DX)=0 THEN P%(DY,DX)=U*K               <FVN>
450 IF H>127 THEN H=H-128                                     <H81>
455 IF H>63 THEN H=H-64                                       <H36>
460 IF H>31 THEN H=H-32                                       <KV1>
465 IF H>15 THEN H=H-16                                       <KB3>
470 IF H>7 AND P%(DY,DX+1)=0 THEN P%(DY,DX+1)=U*K             <59R>
475 NEXT C                                                    <6KH>
480 NEXT B                                                    <AT1>
485 NEXT A                                                    <UJG>
490 IF K=S THEN 505                                           <DCL>
495 K=K+1                                                     <SNP>
500 GOTO 390                                                  <PEB>
505 FOR A=0 TO 99                                             <PGQ>
510 FOR B=0 TO 79                                             <1JK>
515 IF P%(A,B)=0 THEN P%(A,B)=U*(K+1)                         <DJ9>
520 NEXT B                                                    <AL1>
525 NEXT A                                                    <URG>
530 SLOW : GRAPHIC 0,1                                        <VJT>
535 PRINT : PRINT : PRINT : PRINT                             <HDF>
540 PRINT "MOECHTEN SIE DIE GRAFIK"                           <P4P>
```

```
545 PRINT "{DOWN}NIEDRIGAUFLOESEND ( N )"                <QCP>
550 PRINT "{DOWN}ODER"                                   <7BK>
555 PRINT "{DOWN}HOCHAUFLOESEND ( H )";                  <QKM>
560 INPUT GR$                                            <RTM>
565 IF GR$="H" THEN 700: ELSE 600                        <B51>
600 GRAPHIC 3,1                                          <D2N>
605 SCALE 0                                              <NI9>
610 FOR Y=0 TO 99                         : REM ZE
    ICHNEN DES RELIEFS                                   <UT8>
615 YC=Y+50                               : REM NI
    EDRIGE AUFLOESUNG                                    <2QJ>
620 FOR X=0 TO 79                                        <K9K>
625 XA=50-Y/2                                            <C40>
630 XB=INT(XA)                                           <0F4>
635 XC=XB+X                                              <LNR>
640 YD=YC-P%(Y,X)                                        <SLB>
645 DRAW 1,XC,YC TO XC,YD+1                              <7DN>
650 DRAW 3,XC,YD                                         <RCD>
655 DRAW 2,(XC+1),(YC-1) TO (XC+1),YD                    <6SJ>
660 NEXT X                                               <PVC>
665 NEXT Y                                               <DPS>
670 GOTO 800                                             <UIK>
700 GRAPHIC 1,1: COLOR 1,2                               <T6G>
705 SCALE 0                                              <NM9>
710 FOR Y=0 TO 99 STEP 2                  : REM ZE
    ICHNEN DES RELIEFS                                   <KTJ>
715 YC=Y+50                               : REM HO
    HE AUFLOESUNG                                        <2AJ>
720 FOR X=1 TO 79                                        <K9S>
725 XA=50-Y/2                                            <D40>
730 XB=INT(XA)                                           <2F4>
735 XC=XB+X                                              <LVR>
740 YD=YC-P%(Y,X): YE=YC-P%(Y,X-1)                       <QG5>
745 DRAW 0,2*XC,YC TO 2*XC,YD+1                          <1E7>
750 DRAW 0,2*XC-1,YC TO 2*XC-1,YD+1                      <SJE>
755 DRAW 1,2*XC-2,YE TO 2*XC,YD                          <101>
760 NEXT X                                               <9TC>
765 NEXT Y                                               <TRS>
800 GRAPHIC 0,1                                          <GIH>
805 PRINT : PRINT "NAME ZUM ABSPEICHERN DERGRAFIK
    "                                                    <IM1>
810 INPUT "MAX.13 ZEICHEN";NA$                           <VFG>
815 IF GR$="H" THEN GRAPHIC 1: ELSE GRAPHIC 3            <7HR>
820 BSAVE(NA$),ON B0,P8192 TO P16383          : RE
    M SICHERN DER FERTIGEN GRAFIK                        <A07>
825 BSAVE(NA$)+"/C", ON B0,P7168 TO P8191     : RE
    M UND DER ENTSPR.FARBEN                              <CJB>
830 FOR Z=1 TO S                                         <IM2>
835 N$="HOEHE"+STR$(Z)                        : RE
    M LOESCHEN DER ARBEITSDATEIEN                        <KCC>
840 SCRATCH(N$)                                          <3HU>
845 NEXT Z                                               <TBT>
850 GRAPHIC 0,1                                          <LIH>
855 PRINT : PRINT : PRINT : PRINT                        <ITF>
860 INPUT "MOECHTEN SIE EINE HARDCOPY (J/N)";X$          <GHR>
865 IF X$="N" THEN 885                                   <IB3>
870 PRINT : INPUT "NORMAL(1) ODER VERGROESSERT(2)
    ";X$                                                 <RN2>
875 IF X$="1" THEN GOSUB 1500                            <SK3>
880 IF X$="2" THEN GOSUB 2000                            <99L>
885 SLOW                                                 <70O>
890 IF GR$="H" THEN GRAPHIC 1: ELSE GRAPHIC 3            <D0H>
895 END                                                  <U3Q>
1000 PRINT "{2UP,RVSON}LINIE AN ANDEREM PUNKT FOR
     TSETZEN:P" : REM HILFEN ANZEIGEN                    <IP2>
1005 PRINT "{RVSON}NAECHSTE LINIE:N{3SPACE}BEENDE
     N:Q{7SPACE}"                                        <1LA>
1010 PRINT : PRINT                                       <FCQ>
1015 RETURN                                              <GJ7>
1100 INPUT "{UP}X-KOORDINATE:";X
     : REM AUSGANGSPUNKT EINER LINIE                     <PFF>
1105 INPUT "{UP,21RIGHT}Y-KOORDINATE:";Y                 <N83>
1110 LOCATE X,Y                                          <49L>
1115 RETURN                                              <GN7>
1200 PRINT "{UP,39SPACE}"                                <95P>
1205 INPUT "{UP}X-KOORDINATE:";X$             : RE
     M NAECHSTER PUNKT EINER LINIE                       <GES>
1210 IF X$="P" OR X$="N" OR X$="Q" THEN 1235             <QJC>
1215 INPUT "{UP,20RIGHT}Y-KOORDINATE:";Y$                <L9E>
1220 XA=VAL(X$): YA=VAL(Y$)                              <K68>
1225 DRAW 2 TO XA,YA                         : REM L
     INIE ZEICHNEN                                       <2T4>
1230 GOTO 1200                                           <10G>
1235 IF X$="P" THEN 1240: ELSE IF X$="N" THEN 125
     5: ELSE 300                                         <P0G>
1240 PRINT "{UP,39SPACE}"                                <9LR>
1245 GOSUB 1100                              : REM AB
     SETZEN IN DER GLEICHEN                              <0M3>
1250 GOTO 1200                               : REM HO
     EHENSTUFE                                           <69S>
1255 LOCATE 0,0                              : REM AU
     SGANGSPUNKT FUER PAINT                              <TDV>
1260 GRAPHIC 3                                           <QNO>
1265 PAINT 2,1,1                             : REM AL
     LE BILDPUNKTE UEBER DER                             <JNK>
1270 :                                       : REM AK
     TUELLEN HOEHENLINIE                                 <68R>
1275 :                                       : REM WE
     RDEN NICHT VON PAINT                                <40H>
1280 :                                       : REM BE
     TROFFEN                                             <997>
1285 S$=STR$(S): N$="HOEHE"+S$                           <M0B>
1290 BSAVE(N$),ON B0,P08192 TO P16383        : REM AR
     BEITSKOPIE DES BILDSCHIRMS                          <FES>
1295 S=S+1                                               <609>
1300 GRAPHIC 4,1                                         <0AQ>
1305 PRINT : PRINT "ES FOLGT DIE EINGABE"                <FLL>
1310 GOTO 220                                            <222>
1500 FAST                                    : REM PU
     NKT-FUER-PUNKT-HARDCOPY                             <20R>
1505 DIM D%(320,7)                           : REM DE
     R GRAFIK                                            <RTP>
1510 OPEN 4,4                                            <OHE>
1515 PRINT#4,CHR$(8);                                    <C90>
1520 FOR N=8192 TO 16191 STEP 320                        <UB9>
1525 FOR X=0 TO 7                                        <PIL>
1530 FOR Y=0 TO 319 STEP 8                               <0B4>
1535 A=N+X+Y                                             <4ER>
1540 B=PEEK(A)                                           <DSG>
1545 IF B>=128 THEN D%(Y,V)=1: B=B-128                   <SG4>
1550 IF B>=64 THEN D%(Y+1,V)=1: B=B-64                   <N3P>
1555 IF B>=32 THEN D%(Y+2,V)=1: B=B-32                   <4VV>
1560 IF B>=16 THEN D%(Y+3,V)=1: B=B-16                   <AVV>
1565 IF B>=8 THEN D%(Y+4,V)=1: B=B-8                     <U2R>
1570 IF B>=4 THEN D%(Y+5,V)=1: B=B-4                     <9KT>
1575 IF B>=2 THEN D%(Y+6,V)=1: B=B-2                     <NFU>
1580 IF B=1 THEN D%(Y+7,V)=1                             <UKA>
1585 NEXT Y                                              <BLS>
1590 V=V+1                                               <A5R>
1595 IF V=7 THEN GOSUB 1620                              <1TC>
1600 NEXT X                                              <FTC>
1605 NEXT N                                              <P7N>
1610 CLOSE 4                                             <SML>
1615 RETURN                                              <GB7>
1620 V=0                                                 <7CP>
1625 FOR H=0 TO 319                                      <23J>
1630 P1=D%(H,V): D%(H,V)=0                               <EBA>
1635 P2=2*D%(H,V+1): D%(H,V+1)=0                         <FC1>
1640 P3=4*D%(H,V+2): D%(H,V+2)=0                         <LUD>
1645 P4=8*D%(H,V+3): D%(H,V+3)=0                         <IU9>
1650 P5=16*D%(H,V+4): D%(H,V+4)=0                        <40J>
1655 P6=32*D%(H.V+5): D%(H.V+5)=0                        <4ON>
1660 P7=64*D%(H,V+6): D%(H,V+6)=0                        <BSR>
1665 P=P1+P2+P3+P4+P5+P6+P7+128                          <JFU>
1670 PRINT#4,CHR$(P);                                    <ORA>
1675 NEXT H                                              <1OK>
1680 PRINT#4,CHR$(128)                                   <V7F>
1685 RETURN                                              <0P7>
2000 DIM D%(320,7): DIM PA%(320): DIM PB%(320): R
     EM VERGOESSERTE HARDCOPY                            <SR5>
2005 FAST                                    : REM D
     ER GRAFIK                                           <64B>
2010 OPEN 4,4                                            <O9E>
2015 PRINT#4,CHR$(8);                                    <DPU>
2020 FOR N=8192 TO 16191 STEP 320                        <UG9>
2025 FOR X=0 TO 7                                        <QIL>
2030 FOR Y=0 TO 319 STEP 8                               <1R2>
2035 A=N+X+Y                                             <4MR>
2040 B=PEEK(A)                                           <ESG>
2045 IF B>=128 THEN D%(Y,V)=1: B=B-128                   <CK4>
2050 IF B>=64 THEN D%(Y+1,V)=1: B=B-64                   <77P>
2055 IF B>=32 THEN D%(Y+2,V)=1: B=B-32                   <KRV>
2060 IF B>=16 THEN D%(Y+3,V)=1: B=B-16                   <QRV>
2065 IF B>=8 THEN D%(Y+4,V)=1: B=B-8                     <I2R>
2070 IF B>=4 THEN D%(Y+5,V)=1: B=B-4                     <5KT>
2075 IF B>=2 THEN D%(Y+6,V)=1: B=B-2                     <RFU>
2080 IF B=1 THEN D%(Y+7,V)=1                             <6NA>
2085 NEXT Y                                              <RJS>
2090 V=V+1                                               <Q3R>
2095 IF V=7 THEN GOSUB 2120                              <7DM>
2100 NEXT X                                              <VVC>
2105 NEXT N                                              <95N>
2110 PRINT#4,CHR$(15): CLOSE 4                           <3EH>
2115 RETURN                                              <GF7>
2120 V=0                                                 <3C9>
2125 FOR H=0 TO 319                                      <63J>
2130 P1=D%(H,V): D%(H,V)=0                               <69A>
2135 P2=D%(H,V+1): D%(H,V+1)=0                           <MDJ>
2140 P3=D%(H,V+2): D%(H,V+2)=0                           <G0V>
2145 P4=D%(H,V+3): D%(H,V+3)=0                           <DJR>
2150 P5=D%(H,V+4): D%(H,V+4)=0                           <UE7>
2155 P6=D%(H,V+5): D%(H,V+5)=0                           <3P3>
2160 P7=D%(H,V+6): D%(H,V+6)=0                           <4KF>
2165 PA%(H)=3*P1+12*P2+48*P3+64*P4+128                   <CFJ>
2170 PB%(H)=P4+6*P5+24*P6+96*P7+128                      <PTF>
2175 NEXT H                                              <HQK>
2180 FOR H=0 TO 239                                      <N3J>
2185 P=PA%(H)                                            <B5G>
2190 PRINT#4,CHR$(P);                                    <RR6>
2195 PRINT#4,CHR$(P);                                    <RF4>
2200 NEXT                                                <2C0>
2205 FOR H=0 TO 239                                      <E3J>
2210 P=PB%(H)                                            <E98>
2215 PRINT#4,CHR$(P);                                    <DVA>
2220 PRINT#4,CHR$(P);                                    <DJB>
2225 NEXT                                                <0C8>
2230 RETURN                                              <D27>
```

Listing 1. »3-D-Landkarten«. Basic-Listing für den C128. Bitte mit dem Checksummer 128 (Seite 158) eingeben.

Basic total

Eine sehr leistungsfähige Basic-Erweiterung stellen wir Ihnen mit »Supra Basic« vor. Sie stellt unter anderem mächtige Grafikbefehle für den 80-Zeichen-Modus des C 128 zur Verfügung und zeichnet sich durch »Reset-Festigkeit« aus.

Unter Supra Basic haben Sie über hundert neue Befehle zur Verfügung, die das an sich schon leistungsfähige Basic 7.0 des C 128 noch weiter ausbauen. So bietet Supra Basic komfortable Befehle zur Variablenmanipulation, ermöglicht auf einfache Weise Speicheroperationen, die sonst den Assemblerprogrammierern vorbehalten sind und unterstützt vor allem die Grafik des VDC.

Variablenbefehle

Alle verwendeten Befehle werden im Text groß geschrieben, Parameter klein. Parameter, die wahlweise angegeben werden können, stehen in eckigen Klammern (»[« und »]«). Muß von mehreren Parametern mindestens einer angegeben werden, so sind diese durch Schrägstriche (»/«) getrennt.

CLEAR var
Löscht die angegebene Variable und hinterläßt sie in einem definierten Zustand (0 oder Leer-String).

String1 = CUT (string2)
Entfernt alle Space- und SHIFT-Space-Zeichen am **Ende** eines Strings.

String1 = LCUT (String2)
Entfernt alle Space- und SHIFT-Space-Zeichen am **Anfang** eines Strings.

CLFI. Feldvar(1), Feldvar(n)
Löscht alle Variablen des angegebenen Feldbereichs. Ohne FOR-NEXT-Schleife und schneller als LET.

String1 = DEC (String2)
Entfernt das letzte Zeichen von String2.

DUMP
Zeigt alle benutzten Variablen an. Feldvariablen werden mit Dimensionen ausgegeben.

String = DUP (Stringkonstante/Stringvar/Stringverkettung, Multiplikator)
DUP ist eine Anweisung zum Vervielfachen eines Strings (z. B.: y$ = DUP("TEST",2) ergibt »TESTTEST« in y$). Der Multiplikator darf Werte zwischen 1 und 255 annehmen. Wird durch die Duplizierung eine Stringlänge von 255 Byte überschritten, so erfolgt ein »string too long error«.

EXCHANGE Var1, Var2
Tauscht den Inhalt von zwei Variablen aus, ohne dabei den Umweg über eine dritte Variable (Dummy-Variable) zu gehen.

EXFI. Feldvar1(1), Feldvar1(n), Feldvar2(m)
Vertauscht den Inhalt des Bereiches von Feldvar1(1) bis zum letzten angegebenen Index (Feldvar(n)) derselben Variable (z.B. a$(1),a$(10)) mit der neuen Variablen (Feldvar2) ab dem angegebenen Index (m) aus, z.B.:

```
EXFI. a$(1),a$(10),b$(12)
```

INSERT (String1, Position, String2)
Fügt String1 ab der angegebenen Position in String2 ein.

TRANSFER var1,var2
Überträgt den Inhalt von »var1« in »var2« und löscht »var1«.

TRANFI. Feldvar1(1), Feldvar1(n), Feldvar2
TRANFI. entspricht in etwa dem EXFI.-Befehl. Im Unterschied zu diesem löscht er allerdings die Ausgangsvariable (Feldvar1).

ACHTUNG:
CLEAR, CLFI., EXCHANGE,EXFI., TRANSFER und TRANFI. arbeiten nur bei Variablen des gleichen Typs korrekt!
Vorsicht sollten Sie walten lassen, wenn Sie TRANFI. oder EXFI. bei ein und derselben Feldvariablen anwenden: »TRANFI. a(12), a(24), a(28)« ist grundsätzlich erlaubt, kann aber zu unangenehmen Fehlern bis hin zu Datenverlusten führen, wenn der Zielbereich (Feldvar2) ganz oder teilweise im Quellbereich liegt.

VDC- und Grafik-Befehle

Sämtliche Grafik-Befehle beziehen sich – entgegen vielen anderen Basic-Erweiterungen – auf den 80-Zeichen-Modus (VDC-Chip), da der Speicherbereich für den 40-Zeichen-Modus bereits vom Programm selbst belegt ist.

Alle Befehle, denen als Parameter eine Farbe übergeben wird, verwenden folgende Tabelle für die Farbnummern:
0 = Punkte löschen,
1 = Punkte in Vordergrundfarbe setzen,
2 = Punkte invertieren.

Die Syntax der Befehle für die VDC-Grafik stimmt im wesentlichen mit den entsprechenden Befehlen für die VIC-Grafik überein.

GRAPHIC 1
Schaltet die VDC-Grafik ein.

GRCLR
Löscht den Bildschirmspeicher für die VDC-Grafik

GRON
Schaltet die VDC-Grafik ein und löscht den Bildschirm.

GROFF
Schaltet die VDC-Grafik aus, kopiert den Zeichensatz in das VDC-RAM und löscht den Bildschirm.

VCOLOR farbe,fading
»farbe« ändert die Zeichenfarbe der Grafik. »fading« gibt eine Farbverschiebung an. Beide Parameter dürfen nur Werte zwischen 1 und 16 annehmen.

VLOCATE x,y
Positioniert den unsichtbaren Grafik-Cursor an die angegebene Stelle.

VDRAW farbe,x,y
Setzt einen Punkt in der angegebenen Farbe

VDRAW farbe,x1,y1 to x2,y2 [to x3,y3 [to xn,yn]]
Zieht eine Linie von Position x1,y1 nach x2,y2 und – falls angegeben – von dort weiter nach x3,y3 usw.

VDRAW to x,y
Zieht eine Linie von der aktuellen Cursor-Position zum angegebenen Koordinatenpunkt.

VDRAW farbe,x,y to Länge;Winkel
Zieht eine Linie vom angegebenen Punkt mit der angegebenen Länge und dem angegebenen Winkel (im Bogenmaß).

VBOX farbe,x1,y1,x2,y2[,drehwinkel][,ausfüllen]
Zeichnet ein Rechteck mit der linken oberen Ecke x1,y1 und der rechten unteren Ecke x2,y2. **Vorsicht**: Der Parameter »ausfüllen« kann nicht im Zusammenhang mit der Farbe 2 (invertieren) verwendet werden.

VCIRCLE farbe,x,y, x-radius [,y-radius] [,anfangswinkel] [,endwinkel] [,drehwinkel] [,gradzahl]
Zeichnet einen Kreis oder eine Ellipse (entsprechend den Werten in »x-radius« und »y-radius«) mit dem Mittelpunkt an der Position x,y und dem Radius x-radius. Wird für y-radius

kein Wert angegeben, so übernimmt Supra Basic den Wert von x-radius. Dabei sollten Sie jedoch beachten, daß der C 128 in x-Richtung eine andere Auflösung hat (640 Punkte) als in y-Richtung (200 Punkte).

VPAINT farbe, x,y

Mit der Farbe 2 wird die ganze 80-Zeichen-Grafik invertiert. Die durch x und y markierte Position ist der Startwert für das Ausfüllen der Fläche.

VCHAR farbe,x,y, String

Gibt die Zeichenkette ab dem angegebenen Punkt auf dem Grafik-Bildschirm aus. Der Startpunkt gibt die linke obere Ecke des ersten Zeichens an. Diese Adresse ist im Gegensatz zum original CHAR-Befehl pixelweise veränderbar, also x-Achse 0 bis 639 und y-Achse 0 bis 199 als Parameter. Ist für »farbe« eine 0 angegeben, so wird die Zeichenkette revers ausgegeben. Ist sie 1, so wird die Zeichenkette in normaler Schrift ausgegeben. Bei farbe = 2 wird die Zeichenmatrix mit dem Grafik-Bildschirm Exklusiv-Oder-Verknüpft (invertiert).

VDCSAVE [Adresse]

Speichert das gesamte VDC-RAM in den reservierten Bereich ab $BF00. Von dort kann es mit

```
BSAVE name, on b0, p48896 to p65280
```

auf Diskette gespeichert werden. Mit

```
BLOAD name, on b0
```

holt man es in den Speicher zurück. Sollte eine Adresse angegeben sein, so wird der Inhalt des VDC-RAM ab der angegebenen Adresse abgelegt.

VDCLOAD [Adresse]

Kopiert den RAM-Bereich von $BF00 bis $FEFF ins VDC-RAM. Falls eine Adresse angegeben wurde, wird der Inhalt des Speichers ab dieser Adresse in das VDC-RAM kopiert.

ZKR x,y

Zeichnet an die Position x,y ein Zielkreuz mit der Farbe 2 (invers).

VRDOT farbe

Entspricht dem RDOT-Befehl unter Basic 7.0.

SCREEN speicherstelle,wert

Schreibt an die angegebene Speicherstelle des VDC-RAM den angegebenen Wert.

x = INCSR (speicherstelle)

Gibt den Wert aus, der an der angegebenen Speicherstelle steht. Dabei muß »speicherstelle« ein Hex-Wert sein, also entweder mit einem Buchstaben anfangen oder als Kennzeichnung ein »h« als erstes Zeichen haben.

WVREG registernummer,registerinhalt

Beschreibt das Register mit dem neuen Inhalt.

x = RVREG(registernummer)

Gibt den Wert des Registers aus.

STRVDC adresse,string

Schreibt den String ab der angegebenen Stelle in das VDC-RAM.

XSTVDC adresse,string

Schreibt den String ab der angegebenen Stelle in das VDC-RAM. Dabei wird vorher eine Wandlung der ASCII-Zeichen in Bildschirmcode vorgenommen.

x = VDCSTR (adresse,anzahl)

Erzeugt einen String mit der Länge »anzahl« und füllt ihn mit den Zeichen, die ab der angegebenen Adresse im VDC stehen.

PAGE [Seitennummer]

Ohne Parameter schaltet PAGE auf die jeweils andere Textseite um. Page 1 schaltet Seite 1 und PAGE 2 schaltet Seite 2 ein. Die Taste <ALT> entspricht dem PAGE-Befehl. Die Tasten <ALT SHIFT> entsprechen PAGE 1. Die Tasten <ALT CBM> entsprechen PAGE 2. Ist die Grafik eingeschaltet, oder der aktuelle Ausgabe-Bildschirm im 40-Zeichen-Modus, dann erfolgt durch den PAGE-Befehl oder die entsprechenden Tasten keine Reaktion.

Speicheroperationen

SETBOT bank,adresse

Setzt die Untergrenze des Programm- (Bank 0) oder des Variablen-Speichers (Bank 1) auf die angegebene Adresse. Das Programm wird verschoben und geht so nicht verloren. Ist nicht genügend Speicherplatz für die Verschiebung frei, so erhält man einen »out of memory error«. Ist die Adresse in Bank 0 kleiner 16385 oder die Adresse in Bank 1 kleiner 8192 so gibt der Computer einen »illegal quantity error« aus. Bei SETBOT in Bank 1 werden alle Variablen gelöscht.

SETTOP bank,adresse

Setzt die Obergrenze des Programm- (Bank0) oder des Variablenspeichers (Bank1) neu. Wird eine Obergrenze des Speichers größer 65279 gewählt, so gibt die Basic-Erweiterung einen »illegal quantity error« aus. Wird in der Programm-Bank eine Obergrenze von 48895 überschritten, so muß der Programmierer selbst darauf achten, daß der VDCSAVE-Befehl nicht sein Programm zerstört. Wird eine Obergrenze kleiner dem Ende des Basic-Programms (Bank0) oder kleiner 8192 (Bank1) gewählt, so wird ein »out of memory error« ausgegeben. SETTOP in Bank1 löscht alle Variablen.

RBOT (wert)

Liefert die Untergrenze des Basic-Speichers in Bank 0 oder 1 abhängig vom angegebenen Parameter.

RTOP (wert)

Liefert die Obergrenze des Basic-Speichers in Bank 0 oder 1 abhängig vom angegebenen Parameter.

BASIC (stringausdruck)

Wandelt den im String enthaltenen Ausdruck in eine leserliche Form um, wie dies auch beim LIST-Befehl geschieht. Wird dabei eine Länge von 255 Zeichen überschritten, so erhält man einen »string too long error«.

BASLN (adresse)

Dieser Befehl decodiert eine Basic-Zeile, die an der angegebenen Adresse (in Bank 0) beginnt, und legt diese in einem String ab. Falls sich durch die Decodierung eine Länge ergibt, die größer als 255 Byte ist, so wird ein »string too long error« ausgegeben.

STRMEM adresse,string

Schreibt die ASCII-Werte von »string« ab der angegebenen Adresse in die aktuelle Bank.

string$ = MEMSTR (adresse,länge)

Dieser Befehl holt von der angegebenen Adresse der aktuellen Bank so viele Zeichen, wie durch den Parameter Länge vorgegeben sind, und legt sie in einem String ab.

MOVE quellbank,zielbank,anfadresse,endadresse,zieladresse

Überträgt einen Speicherbereich von »anfadresse« bis »endadresse« in »quellbank« in den Speicher ab »zieladresse« in der »zielbank«.

MEMSWAP quellbank,zielbank,anfadresse,endadresse,zieladresse

Tauscht einen Speicherbereich von »anfadresse« bis »endadresse« in der »quellbank« mit dem Bereich in der »zielbank« ab »zieladresse« aus.

FILL bank,füllwert,anfadresse,endadresse

Belegt den angegebenen Speicherbereich in der angegebenen Bank mit einem Füllwert.

DPOKE adresse,wert

Dient zum Poken eines 16-Bit-Werts als 2 Byte im Low/High-Format des Betriebssystems. Hiermit kann auch der Interruptvektor verbogen werden.

x = DPEEK (adresse)

Liefert einen 16-Bit-Wert von 2 Byte, die im Low/High-Format ab der angegebenen Adresse im Speicher stehen.

Editierbefehle

HIDE [zeilennummer]
Versteckt das Basic-Programm bis zur angegebenen Zeilennummer. Ohne »zeilennummer« wird das gesamte Programm versteckt.
UNITE
Bringt den versteckten Basic-Teil wieder zum Vorschein.
MERGE programmname,8
Bewirkt das Zuladen eines zweiten Basic-Programmes zum bereits im Speicher vorhandenen. Die Zeilennummern können nun mit RENUMBER verändert werden; sie müssen größer sein als die des ersten Programms.
MERGE 2
Verknüpft die beiden Programme, so daß der erste Teil wieder zu sehen und zu nutzen ist.
FIND text/befehl
Listet alle Zeilen, in denen »text« oder »befehl« vorkommt. Texte, die im Programm in Anführungszeichen stehen, müssen auch hier in Anführungszeichen stehen, da Supra Basic sonst nicht zwischen Befehl und Text unterscheiden kann und es daher zu Fehlern kommt. Nach dem FIND-Befehl darf kein weiterer Befehl in der Basic-Zeile stehen, da dieser von der FIND-Routine als Suchobjekt angesehen würde (außer bei Text in Anführungszeichen, wo nach dem zweiten Anführungszeichen in der Basic-Zeile fortgefahren wird).
CLIST
Modifiziert die List-Routine. Beim Listen werden alle Befehle von Supra Basic und alle REM-Zeilen invers dargestellt. Die formatzerstörenden Zeichen RETURN (chr$(13)), SHIFT-RETURN (chr$(141)), ESC (chr$(27)) und DEL(chr$(20)) werden als chr$-Code in den Text eingefügt. Außerdem wird mit dem Befehl WKEY (s.u.) auf einen Tastendruck gewartet. Durch ein erneutes CLIST wird die Modifizierung aufgehoben.
WKEY
Veranlaßt den CLIST-Befehl, nach jeder angezeigten Seite auf einen Tastendruck zu warten.
OLD
Gegenbefehl zu NEW. Funktioniert nur, wenn seit dem NEW die Zeiger auf den Programmanfang nicht verändert und kein neues Programm eingegeben wurde.
OLDKEY
Durch diesen Befehl wird die Belegung der Funktionstasten, wie sie beim Einschalten vorliegt, wiederhergestellt.
RELOCATE anfzeile, endzeile, zielzeile [,startzeile_neu] [,Schrittweite]
Verschiebt den Programmbereich von »anfzeile« bis einschließlich »endzeile« vor die »zielzeile«. Um den Bereich an das Programmende zu verschieben, müssen Sie bei »zielzeile« eine Nummer eingeben, die höher ist als die der bisherigen letzten Zeile. Wenn »endzeile« kleiner als »anfzeile« ist, oder »zielzeile« in dem zu verschiebenden Bereich liegt, so gibt dies einen »illegal quantity error«. Nach dem Verschieben wird automatisch eine Neunumerierung aller Programmzeilen durchgeführt. Dabei beträgt der Startwert und die Schrittweite jeweils 10. Die beiden letzten (wahlweisen) Parameter bestimmen, mit welcher Zeilennummer das Programm beginnen soll, und in welcher Schrittweite es durchnumeriert werden soll (falls ein anderer Wert als 10 gewünscht wird).
EXECUTE string
Führt den in »string« enthaltenen Basic-Befehl aus. Ist am Anfang des Strings eine Zeilennummer, so wird der String als Basic-Zeile in das Programm eingefügt. Das Programm darf jedoch in keinem Unterprogramm (GOSUB) und keiner Schleife (FOR...NEXT und DO...LOOP) verändert werden, da dadurch ein Absturz provoziert oder das Programm zerstört werden kann. Sie sollten auch nicht versuchen, die Zeile zu ändern, in der der EXECUTE-Befehl steht, da Sie sonst einen Fehler verursachen.

Echtzeituhr

SETTIME string
Setzt die in dieser Basic-Erweiterung integrierte Echtzeituhr und startet sie. Der String muß sieben Zeichen lang sein und hat folgende Struktur:
hhmmssz (mit hh = Stunden; mm = Minuten; ss = Sekunden; z = Zehntelsekunden).
Die Echtzeituhr benutzt den CIA 1 zum Timing.
TIME (dummy)
Gibt die Uhrzeit in folgendem Format als String aus :
hh:mm:ss.z

Sonstige Befehle

XKEY (nummer,text)
Entspricht dem KEY-Befehl, nur sind die Tastennummern um 1 kleiner und es können alle zehn Funktionstasten verändert werden.
REPEAT
Schaltet die gewohnte Tastenwiederholfunktion ein, die auch nach dem Einschalten aktiv ist.
REPEATON
Durch diesen Befehl werden nur noch SPACE, die Cursortasten, INST und DEL durch die Tastenwiederholfunktion unterstützt.
REPEATOFF
Schaltet die Wiederholfunktion vollständig ab.
CUON
Schaltet im Programmodus den Cursor ein (dies ist beispielsweise nützlich für Eingabeschleifen mit GET- oder GETKEY-Befehlen.)
COFF
schaltet den Cursor aus.
LOCKS
Schaltet die STOP-Taste und <STOP RESTORE> ab.
UNLOCKS
Schaltet die STOP-Taste und <STOP RESTORE> wieder an.
DINLOCK
Schaltet auf den DIN-Zeichensatz um. Dieser kann anschließend nicht mehr auf ASCII zurückgeschaltet werden.
DINFREE
Gibt die Umschaltung von ASCII auf DIN wieder frei.
WRUSR adresse
Setzt den USR-Vektor auf »adresse«.
LLIST [anfang ende]
Listet das aktuelle Listing auf dem Drucker. Voreingestellt ist Geräteadresse 4, Sekundäradresse 7. Mit »anfang« und »ende« kann ein Teilbereich des Programms angegeben werden.
LSET geräteadresse,sekundäradresse
Setzt Geräteadresse und Sekundäradresse des Druckers für LLIST und LPRINT.
LPRINT ausdruck
Funktioniert wie der PRINT-Befehl, nur daß die Ausgabe auf den Drucker umgeleitet wird. Mit LSET können die Adressen eingestellt werden (voreingestellt sind 4 und 7).
PAUSE [parameterliste,]wartezeit
Wartet solange wie in »wartezeit« angegeben mit der Programmausführung. Wenn eine Parameterliste angegeben wird, muß der erste Ausdruck der Liste ein String sein. In der Parameterliste darf kein Komma vorkommen.

CENTRE parameterliste
Gibt die Parameterliste zentriert in der aktuellen Zeile aus. Ist der Inhalt der Parameterliste größer als die Anzahl der Spalten in der aktuellen Zeile, so erhält man einen »string too long error«.
AT (spalte,zeile)
Positioniert den Cursor an die angegebene Stelle auf dem Bildschirm. Hierdurch kann auch eine Ausgabe außerhalb des aktuellen Windows erfolgen (aber natürlich nur innerhalb des Bildschirms), allerdings dürfen dann in der folgenden Parameterliste (des PRINT- oder CENTRE-Befehls) keine Cursor-Steuerzeichen, kein DEL, kein INST, kein CLR und kein HOME vorkommen, weil deren Funktionen sich nur auf das aktuelle Fenster beziehen können.
DISK parameterliste
Sendet die Parameterliste auf dem Befehlskanal an die Floppy. Voreingestellt ist Geräteadresse 8.
DISET geräteadresse
Mit diesem Befehl wird die voreingestellte Geräteadresse (z.B. für DISK) auf einen Wert von 8 bis 11 geändert.Bei anderen Werten wird eine Fehlermeldung angezeigt.
TKEY
Wartet auf einen Tastendruck.
OFINPUT [text;parameterliste,]variablenliste
Ein modifizierter INPUT-Befehl. Die Parameterliste ermöglicht das Ausgeben von Werten ähnlich wie mit dem PRINT-Befehl, allerdings können keine Kommata verwendet werden, da diese das Trennzeichen zur Variablenliste darstellen. Text und Parameterliste sind wie beim Original-INPUT-Befehl optional. Dies bedeutet, daß, wenn am Anfang kein Text in Anführungszeichen steht, angenommen wird, daß gleich hinter OFINPUT die Variablenliste beginnt. Bei Einlesen nur einer (!) Stringvariablen werden alle Zeichen eingelesen (also auch Komma, Anführungszeichen usw.). Die Tasten CRSR-aufwärts und CRSR-abwärts sind in ihrer Funktion blockiert. CRSR-rechts und CRSR-links funktionieren, aber Sie können mit diesen Tasten das Eingabefeld von maximal 160 Zeichen nicht verlassen. Das gleiche gilt für die Tasten INST und DEL. Durch HOME kommen Sie an den Anfang des Eingabefeldes. Durch CLR wird das Eingabefeld gelöscht und der Cursor steht am Anfang des Eingabefeldes.
INPUTFORM spalte, zeile, länge, kommentar, vorgabe, zugelassene_zeichen, zielvariable
Gibt an der angegebenen Stelle auf dem Bildschirm einen Kommentar aus. Dahinter wird die Vorgabe ausgegeben. Kommentar und Vorgabe können Parameterlisten wie beim PRINT-Befehl sein, nur dürfen hier keine Kommata vorkommen, da diese die Trennzeichen zum nächsten Abschnitt bilden. »zugelassene_zeichen« gibt an, welche Tasten außer <SPACE>, <CRSR-rechts>, <CRSR-links>, <HOME>, <CLR>, <INST>, <DEL> und <RETURN> zugelassen sind. Die Eingabe wird in »zielvariable« abgelegt. Das Eingabefeld beginnt mit dem Start der »vorgabe« und kann zwischen 1 und 160 Zeichen lang sein. Ansonsten verhält sich INPUTFORM wie OFINPUT.
DINPUT (abschlußzeichen,länge,filenummer)
Liest aus dem geöffneten File einen String mit der angegebenen Länge, dabei werden alle Zeichen eingelesen (einschließlich chr$(0), chr$(13) ....). Die Länge darf maximal 255 Byte betragen. Wahlweise kann auch auf ein bestimmtes Byte hin die Eingabe vor Erreichen des letzten einzulesenden Zeichens beendet werden. Das Abschlußzeichen wird nicht im String abgelegt. Um dies zu erreichen, muß der ASCII-Wert dieses Zeichens als »abschlußzeichen« angegeben werden. Soll auf jeden Fall die gesamte Länge eingelesen werden, so muß der Wert für das Abschlußzeichen zwischen 256 und 65535 liegen.
RESET
Führt einen Warmstart durch.
LWIND [zeile_1,[zeile_n]]
Ohne Parameter wird das Bildschirmfenster gleich der maximalen Größe. Mit nur einem Parameter wird diese Zeile zum aktuellen Bildschirmfenster. Mit beiden Parametern wird das Bildschirmfenster von der ersten bis zur letzten angegebenen Zeile neu definiert.
CGOTO zeilennummer/label
Springt an die angegebene Stelle. Dabei kann die Zeilennummer jeden Wert zwischen 0 und 65535 annehmen. Der Befehl CGOTO wird nicht durch RENUMBER angepaßt. Die Zeilennummer kann auch als Variable oder Formel angegeben sein, beispielsweise:
```
CGOTO a*2+h-3
```
»CGOTO label« dagegen springt in die Zeile, in der ein PROC-Befehl mit dem angegebenen Label steht. Es wird in der Zeile nach dem PROC-Befehl mit dem Programmablauf fortgefahren.
CGOSUB zeilennummer
Verhält sich im wesentlichen wie das original GOSUB. Zusätzlich gelten die Regeln von CGOTO.
PROC"label"
Setzt eine Sprungmarke für CGOTO und CGOSUB. Die Programmausführung wird in der Zeile nach dem PROC-Befehl fortgesetzt. Trifft das Programm während des normalen Programmablaufs auf einen PROC-Befehl, so werden alle Zeichen hinter PROC bis zum nächsten Doppelpunkt oder dem Zeilenende überlesen. Folglich dürfen im Labelnamen alle Zeichen außer dem Doppelpunkt vorkommen. Zwischen PROC und dem ersten Anführungszeichen darf kein Leerzeichen stehen.
ERASE
Löscht ein Basic-Programm unwiderruflich.
KILL
führt zum Absturz. **Vorsicht!!** KILL kann nicht abgebrochen oder rückgängig gemacht werden.
CODE programmname
Verschlüsselt ein Basic-Programm. Dieser Code ist ohne fundierte Maschinensprachkenntnisse nur sehr schwer zu entschlüsseln. Das codierte Programm kann anschließend mit DSAVE gespeichert werden. Zum sicheren Laden und Starten dieses Programms sollten Sie folgende Befehle eingeben:
```
LOCKS:BLOADname,ONb0,p(DPEEK(45)):DPOKE4624,DPEEK
(174): DECODE programmname: RUN
```
Bitte geben Sie diese Befehle im Direktmodus ein, keinesfalls programmgesteuert (es sei denn durch EXECUTE).
DECODE programmname
Decodiert ein vorher mit CODE verschlüsseltes Basic-Programm.
OFF[char]
Schaltet die Basic-Erweiterung ab, gibt die blockierten Bereiche frei und setzt die alten Vektoren. Das Programm kann nicht wieder gestartet werden. Befindet sich dagegen hinter OFF noch ein weiteres beliebiges Zeichen, so wird die Basic-Erweiterung abgeschaltet, aber die Bereiche nicht freigegeben. Durch Drücken der RESTORE- oder RESET-Taste wird die Basic-Erweiterung wieder aktiviert. Bei einem OFF ohne zusätzliches Zeichen ergibt sich folgender Ablauf:
1. Das Basic-Programm bleibt bei $4000 stehen.
2. Der Zeiger auf den Programmanfang wird auf die Adresse gesetzt, die nach dem Einschalten des Rechners als Programmstart definiert ist ($1c01).
3. Alle Variablen werden gelöscht. Die Common Area wird auf Normalgröße verkleinert.

var1 = USED (var2)
Legt die Programmlänge des aktuellen Basic-Programms in »var1« ab. »var2« wird lediglich als Platzhalter gebraucht.
var = LOW (wert)
Liefert das Low-Byte eines Wertes (entspricht also wert AND 255)
var = HIGH (wert)
Liefert das High-Byte eines Wertes (entspricht also wert /256)
var = RAD (wert)
Konvertiert einen Wert, der im Bogenmaß vorliegt, zu Grad.
var = DEG (wert)
Konvertiert einen Wert, der im Gradmaß vorliegt, ins Bogenmaß.
var1 = RBANK (var2)
Legt die aktuelle Bank in »var1« ab, »var2« hat lediglich Platzhalterfunktion.
var1 = RDLINE (var2)
Legt die letzte bearbeitete Data-Zeile in »var1« ab.
var1 = PRGLINE (var2)
Legt die Nummer der aktuellen Programmzeile in »var1« ab.
var1 = INKEY (var2)
Übergibt »var1« die Nummer der zuletzt gedrückten Funktionstaste:

0	keine Funktionstaste
1-8	<F1> bis <F8>
9	<RUN/STOP>
10	<HELP>

var = EVAL (string)
Wertet die in der Zeichenkette enthaltene Formel aus.
ROUND (stellen,wert)
Rundet »wert« auf eine Genauigkeit entsprechend der Anzahl von »stellen«.

Zeiger von Supra Basic

Adresse	Funktion des Zeigers
$2D/2E	Anfang des Programmspeichers
$2F/30	Anfang des Variablenspeichers
$2FC-315	Basic- und Interrupt-Vektoren
$A00/A01	Basic-Warmstart
$A2E	Beginn des VDC Zeichen-RAM
$A2F	Beginn des Attribut-RAM
$1210/1211	Ende des Basic-Programmes
$1212/1213	Ende des Programm-Speichers

Tabelle 1. Diese Zeiger verwendet Supra Basic intern. Sie sollten möglichst nicht verändert werden.

var = COT (wert)
Legt den Cotangens von »wert« (der einen Winkel im Bogenmaß darstellt) in »var« ab.
var = LG (wert)
Legt den Logarithmus von »wert« zur Basis 10 in »var« ab.
var = FRAC (wert)
Legt die Nachkommastellen von »wert« in »var« ab.
var = ODD (wert)
Gibt 0 aus, wenn »wert« gerade ist und -1, wenn er ungerade ist.
var = MIN wert_1[,wert_2[,...wert_n]]
Gibt den kleinsten Wert einer beliebig langen Werteliste an.
var = MAX wert_1[,wert_2[,...wert_n]]
Gibt den größten Wert einer beliebig langen Werteliste aus.
var = DIV(dividend,divisor)
Legt das Ergebnis der Division »dividend« /»divisor« als Ganzzahl ohne Nachkommastellen in die Variable »var«.
var = MOD(dividend,divisor)
Legt den Rest der Division »dividend« /»divisor« in die Variable »var«.
var =$hexzahl
Wandelt die folgende Hexadezimalzahl in eine Dezimalzahl um. Die Hexadezimalzahl darf maximal 4 Stellen lang sein.
var = %binzahl
Wandelt die folgende Binärzahl in eine Dezimalzahl um. Die Binärzahl darf maximal 16 Stellen lang sein.
var = &dezzahl/formel
Wandelt den Wert des dezimalen Ausdruckes in einen String, der eine hexadezimale Zahl des gleichen Wertes enthält. Die Stringlänge beträgt zwischen 1 und 4 Zeichen.
var = #dezzahl/formel
Wandelt den Wert des dezimalen Ausdruckes in einen String, der eine Binärzahl des gleichen Wertes enthält. Die Stringlänge beträgt zwischen 1 und 16 Zeichen.

Eingabehinweise

Tippen Sie das Programm (Listing 1) mit dem MSE ab. Denken Sie daran, Supra Basic auf Diskette abzuspeichern, bevor Sie den MSE verlassen. Starten Sie Supra Basic mit:

```
BOOT"Supra Basic +"
```

Nach dem Laden belegt das Programm den Bereich 4864 ($1300) bis 16146 ($3F12). Die genaue Speicherbelegung können Sie Tabelle 2 entnehmen.

Wegen der Common-Area muß in der Bank 1 der Variablen-Beginn auf $2000 hochgesetzt werden. Dabei werden alle Variablen gelöscht. Des weiteren simuliert das Programm bei der Initialisierung den Befehl NEW. Trotzdem kann es von einem Basic-Programm durch

```
GRAPHIC 1:BOOT"Supra Basic +" : OLD
```

nachgeladen werden.

Allerdings darf diese Zeile nur bei aktiviertem Supra Basic eingegeben werden. Die in Tabelle 1 aufgeführten Zeiger sollten niemals direkt verändert werden, um eine einwandfreie Funktion von Supra Basic zu gewährleisten.

Speicherbelegung

$0000-$1FFF	Common Area Bank 0
$2000-$3FFF	Hier liegt der obere Programmteil
$4000-$BEFF	Basic-Programmspeicher
$BF00-$FEFF	Zwischenspeicher für das VDC-RAM
$FF00-$FFFF	Kernel Bereich Bank 1
$2000-$FEFF	Variablenspeicher
$FF00-$FFFF	Kernelbereich

Tabelle 2. So teilt sich der Speicher unter Supra Basic auf

Der Bereich in der Zeropage wird von Supra Basic nur als Zwischen- und Hilfsspeicher für die verschiedenen Routinen genutzt und bleibt ansonsten frei.

Da Supra Basic den Grafik-Speicher für den 40-Zeichen-Bildschirm selbst benötigt, ist dieser Grafik-Modus bei aktivierter Erweiterung nicht ansteuerbar, was bei den Möglichkeiten, die die 80-Zeichen-Grafik nun bietet, wohl leicht zu verschmerzen ist.

Wichtig:
Supra Basic ist reset-fest. Nach der Initialisierung kann es nur durch den Befehl OFF oder Ausschalten des Computers deaktiviert werden. Eine erneute Initialisierung nach OFF ist nicht möglich. (Martin Ilse/so)

```
Name : Supra Basic +      1300 3ff9
-----------------------------------
1300 : 4c 4f 3a a5 30 85 62 a5  3b
1308 : 2f 4c 1b 13 20 92 13 20  aa
1310 : c6 13 a5 61 18 69 07 90  ff
1318 : 02 e6 62 85 61 c5 31 d0  81
1320 : eb a6 62 e4 32 d0 e5 f0  b7
1328 : 3b 20 92 13 a9 28 20 c8  68
1330 : 13 a0 04 20 00 43 85 fa  bf
1338 : d0 03 20 c6 13 20 7a 13  ad
1340 : c6 fa d0 f6 a9 29 20 c8  8c
1348 : 13 20 c6 13 a0 02 20 00  1a
1350 : 43 aa c8 20 00 43 65 62  93
1358 : 85 62 8a 18 65 61 90 02  5c
1360 : e6 62 85 61 c5 33 d0 c1  c2
1368 : a5 62 c5 34 d0 bb 60 48  33
1370 : 20 81 14 68 20 32 8e 4c  c9
1378 : d2 13 a5 fa 0a 69 04 a8  ea
1380 : 20 00 43 aa 88 20 00 43  d7
1388 : ca e0 ff d0 02 e9 00 4c  e4
1390 : 6f 13 a0 01 20 00 43 85  eb
1398 : fc 48 88 20 00 43 85 fb  07
13a0 : 0a 90 08 68 0a 90 30 a9  3b
13a8 : 25 d0 09 68 0a 90 04 a9  0d
13b0 : 24 d0 01 98 85 fd a5 fb  67
13b8 : 20 c8 13 a5 fc 20 c8 13  d0
13c0 : a5 fd f0 13 d0 02 a9 2c  1e
13c8 : 29 7f 48 20 81 14 68 20  62
13d0 : df 90 a2 40 8e 00 ff 60  52
13d8 : 20 f1 1a 20 06 15 e0 01  0a
13e0 : f0 60 b0 03 4c 28 7d a5  d4
13e8 : fc a4 fd 85 66 84 67 20  cf
13f0 : 9d 1a d0 06 20 09 1b 4c  df
13f8 : 3c 14 a9 00 a6 fe 18 65  36
1400 : fb b0 46 ca d0 f8 85 fa  1f
1408 : 20 ac 15 20 88 86 20 a5  50
1410 : 15 a6 fe a0 00 b1 fc 91  f1
1418 : 64 c8 c4 fb d0 f7 98 18  f0
1420 : 65 64 85 64 90 02 e6 65  25
1428 : ca d0 e8 a5 35 85 64 a5  a5
1430 : 36 85 65 20 09 1b 20 81  73
1438 : 14 20 e3 86 20 ac 15 ca  77
1440 : 86 0f 68 68 68 4c 56 1d
1448 : 79 20 81 14 4c ed a5 20  bf
1450 : 5c 79 20 ef 77 a9 2d 8d  03
1458 : 7c 14 a9 1c 8d 7d 14 a9  35
1460 : 3f 8d 77 14 20 dd 77 a4  de
1468 : 67 a5 66 20 9d 1a f0 11  d0
1470 : 88 20 a5 15 b1 fc a2 3f  20
1478 : 8e 00 ff 99 2d 1c 98 d0  f1
1480 : ef a9 00 8d 00 ff 60 ae  d4
1488 : d5 03 48 ad 00 ff 48 20  08
1490 : 81 14 bd f0 f7 aa 68 8d  3a
1498 : 00 ff 68 60 20 ee 1a 20  e0
14a0 : 09 88 a9 00 8d dd 14 8a  85
14a8 : 20 88 86 a8 a2 00 2c dd  5a
14b0 : 14 30 03 20 87 14 8e be  72
14b8 : 14 98 f0 10 88 a9 00 8d  47
14c0 : 00 ff b1 16 20 a5 15 91  96
14c8 : 64 98 d0 f0 20 09 1b ee  5f
14d0 : dd 14 30 03 4c 36 14 20  2b
14d8 : 81 14 4c e3 86 00 86 fb  6d
14e0 : ae f5 14 9d 2d 1b ee f5  95
14e8 : 14 d0 06 20 57 19 4c ed  35
14f0 : a5 a6 fb 18 60 00 20 ef  51
14f8 : 77 a9 2d 8d 7c 14 a9 1d  8a
1500 : 8d 7d 14 4c 5f 14 20 dd  ad
1508 : 77 a5 66 a4 67 85 fc 84  20
1510 : fd 20 09 88 86 fe 60 20  93
1518 : 06 15 20 5c 79 20 ef 77  84
1520 : 20 dd 77 a5 66 a4 67 20  2b
1528 : e0 87 a5 fc a4 fd 20 e0  51
1530 : 87 20 a5 15 a0 02 b1 fc  ae
```

```
1538 : 99 9a 02 b1 66 99 9d 02  83
1540 : 99 97 02 88 10 f0 a0 40  c2
1548 : 8c 00 ff a4 fe 8c 9d 02  37
1550 : ad 97 02 38 e5 fe 8d 97  0c
1558 : 02 a5 fe 18 6d 98 02 90  b4
1560 : 03 ee 99 02 8d 98 02 ad  82
1568 : 9d 02 18 6d 9a 02 b0 2c  8f
1570 : 6d 97 02 b0 27 20 88 86  e2
1578 : a9 9d 85 70 a9 02 85 71  03
1580 : 20 4e 87 a9 9a 85 70 a9  c9
1588 : 02 85 71 20 4e 87 a9 97  a4
1590 : 85 70 a9 02 85 71 20 4e  f9
1598 : 87 4c 36 14 4c ed a5 20  60
15a0 : 3f 4b 4c ac 15 a2 7f d0  33
15a8 : 05 20 af 7a a2 00 8e 00  5d
15b0 : ff 60 20 05 54 4c ac 15  0d
15b8 : 20 a0 86 4c ac 15 a0 00  49
15c0 : a2 00 8e 00 ff a5 65 d0  6a
15c8 : 02 c6 66 c6 65 a5 fc d0  b9
15d0 : 02 c6 fd c6 fc b1 fc a2  24
15d8 : 00 8e 00 ff 2c 4c 16 10  bd
15e0 : 12 48 b1 65 ae c1 15 8e  9a
15e8 : 00 ff 91 fc ae d8 15 8e  0f
15f0 : 00 ff 68 91 65 a5 fc c5  3f
15f8 : fa d0 c5 a5 fd c5 fb d0  20
1600 : bf a9 00 8d 00 ff 60 a0  08
1608 : 00 a9 00 8d 00 ff b1 fa  4b
1610 : a2 00 8e 00 ff 2c 4c 16  15
1618 : 10 12 48 b1 65 ae 0a 16  9a
1620 : 8e 00 ff 91 fa ae 11 16  76
1628 : 8e 00 ff 68 91 65 e6 fa  99
1630 : d0 02 e6 fb e6 65 d0 02  1b
1638 : e6 66 a6 fa e4 fc d0 c9  67
1640 : a6 fb e4 fd d0 c3 a9 00  ae
1648 : 8d 00 ff 60 00 a6 fe 8e  30
1650 : 00 ff a0 00 ad ef 03 91  01
1658 : fa e6 fa d0 02 e6 fb a6  33
1660 : fa e4 fc d0 f2 a6 fb e4  44
1668 : fd d0 ec a9 00 8d 00 ff  aa
1670 : 60 20 a5 15 a4 63 c0 02  59
1678 : d0 20 b1 61 85 65 88 b1  fa
1680 : 61 85 64 88 b1 61 f0 12  dc
1688 : aa 18 65 64 85 64 90 02  e6
1690 : e6 65 8a 91 64 c8 a9 ff  31
1698 : 91 64 a4 63 a9 00 91 61  94
16a0 : 88 10 fb 8d 00 ff 60 a0  a4
16a8 : 7f 8c 00 ff ac e6 03 c0  fd
16b0 : 02 d0 52 b1 61 85 fc 88  2c
16b8 : b1 61 85 fb 88 b1 61 f0  78
16c0 : 22 48 18 65 fb 85 fb 90  b6
16c8 : 02 e6 fc 68 2c 1f 17 10  c2
16d0 : 09 91 fb c8 a9 ff 91 fb  93
16d8 : d0 09 a5 63 91 fb c8 a5  6a
16e0 : 64 91 fb a0 02 b1 63 85  66
16e8 : fc 88 b1 63 85 fb 88 b1  bf
16f0 : 63 f0 12 18 65 fb 85 fb  97
16f8 : 90 02 e6 fc a5 61 91 fb  86
1700 : c8 a5 62 91 fb ac e6 03  2c
1708 : b1 61 aa b1 63 91 61 8a  a8
1710 : 2c 1f 17 10 02 a9 00 91  24
1718 : 63 88 10 ec 4c 81 14 00  82
1720 : 20 aa 21 a5 83 c9 02 8d  3c
1728 : 53 18 d0 03 4c 54 18 a2  29
1730 : 04 20 52 9e 20 f2 9d 20  fd
1738 : dc 21 20 49 18 b0 02 d0  87
1740 : 01 60 20 ea 92 8d 03 ff  78
1748 : a5 33 85 24 a5 34 85 25  c9
1750 : 38 a5 35 e9 03 85 1b a5  f9
1758 : 36 e9 00 85 1c a2 00 86  17
1760 : 63 86 64 ae e3 20 d0 03  7e
1768 : ce e4 20 ce e3 20 20 49  dd
1770 : 18 b0 02 d0 ee ee e3 20  b1
1778 : d0 03 ee e4 20 20 41 18  5a
```

```
1780 : ae e1 20 d0 03 ce e2 20  b3
1788 : ce e1 20 a5 63 20 06 18  83
1790 : 85 63 18 ad e1 20 69 02  4b
1798 : 8d e1 20 90 03 ee e2 20  a3
17a0 : a5 64 20 06 18 85 64 ad  db
17a8 : e1 20 d0 03 ce e2 20 ce  50
17b0 : e1 20 ee e3 20 d0 03 ee  4c
17b8 : e4 20 20 49 18 b0 02 d0  8e
17c0 : bc a2 03 a0 00 a5 25 c5  f0
17c8 : 34 d0 06 a5 24 c5 33 f0  ba
17d0 : 1a a5 24 d0 02 c6 25 c6  58
17d8 : 24 20 b7 03 8d 03 ff 9d  87
17e0 : e1 20 ca 10 ec 20 b5 4b  c3
17e8 : 4c 5d 17 ad e1 20 8d 31  16
17f0 : 11 ad e2 20 8d 32 11 ad  9e
17f8 : e3 20 8d 33 11 ad e4 20  07
1800 : 8d 34 11 4c f2 9d 48 20  f3
1808 : 49 18 b0 18 f0 16 68 d0  8f
1810 : 16 aa a8 a5 25 c5 1c 90  6c
1818 : 0f d0 06 a5 24 c5 1b 90  c4
1820 : 07 4c 3a 4d 68 a9 00 60  1a
1828 : bd e1 20 8d 04 ff 91 24  5e
1830 : 8d 03 ff e6 24 d0 02 e6  ba
1838 : 25 e8 e0 04 d0 ea a9 80  96
1840 : 60 ae 53 18 86 83 4c f9  79
1848 : 20 20 57 25 b0 04 cc 53  f8
1850 : 18 18 60 00 a9 00 8d ec  37
1858 : 20 8d eb 20 20 5d 21 20  ef
1860 : a9 24 49 ff 48 20 5d 21  ab
1868 : 68 20 a1 24 ee ec 20 d0  46
1870 : eb ee eb 20 ad eb 20 c9  1f
1878 : 40 d0 e1 60 20 e5 55 20  6b
1880 : 86 03 f0 4d c9 a3 f0 28  3b
1888 : c9 a6 18 f0 23 c9 2c d0  9b
1890 : 06 20 81 14 4c 80 03 c9  f2
1898 : 3b f0 30 20 ef 77 24 0f  c5
18a0 : 30 da 20 42 8e 20 9a 86  ef
18a8 : 20 e5 55 20 00 56 d0 cf  aa
18b0 : 08 38 20 8d 92 84 0b 20  48
18b8 : f1 87 c9 29 d0 13 28 90  6c
18c0 : 06 8a e5 0b 90 05 aa e8  94
18c8 : ca d0 09 20 80 03 4c 7f  91
18d0 : 18 4c 6c 79 20 00 56 d0  56
18d8 : ef 4c 28 7d 20 e2 18 4c  b9
18e0 : 36 14 20 ee 1a aa d0 f1  24
18e8 : c4 ee b0 ed 8c 0a 19 20  cb
18f0 : 09 88 20 09 1b e0 19 b0  e5
18f8 : e0 20 81 14 85 63 85 64  1e
1900 : 85 65 8a 48 ad 0a 19 4c  0c
1908 : 86 cc 00 a9 00 8d 56 19  22
1910 : a0 ff 20 1e 2d 20 57 55  58
1918 : 20 81 14 20 57 19 c6 fd  57
1920 : 2c 56 19 10 01 60 20 ed  2f
1928 : ff e8 8a e5 fd b0 03 4c  05
1930 : 49 14 4a 65 e6 8d 0a 19  f8
1938 : a5 eb 20 03 19 a9 ff 85  25
1940 : fc a6 fc e8 86 fc e4 fd  cb
1948 : 90 03 f0 01 60 bd 2d 1b  95
1950 : 20 0c 56 4c 41 19 00 a9  c6
1958 : 79 8d 26 03 a9 ef 8d 27  20
1960 : 03 ad f5 14 85 fd 60 48  94
1968 : a9 01 8d 87 03 20 57 19  a7
1970 : 68 4c 3f 4d 20 80 03 20  ca
1978 : f1 1a 20 12 88 a0 00 0a  62
1980 : 48 a9 30 90 02 a9 31 99  20
1988 : 1d 1b 68 c8 c0 08 90 ef  d4
1990 : d0 02 a5 16 c0 10 90 e7  2c
1998 : a2 10 a0 00 a9 1c 85 16  28
19a0 : a9 1b 85 17 e6 16 d0 02  81
19a8 : e6 17 ca f0 06 b1 16 c9  c4
19b0 : 30 f0 f1 a0 fe e8 8c dd  0e
```

Listing 1. »SUPRA BASIC+«

```
19b8 : 14 4c a7 14 20 80 03 c9   04
19c0 : 23 90 04 c9 27 90 06 20   b5
19c8 : c2 3c 4c da 78 a2 00 86   c0
19d0 : 0f 48 20 80 03 20 b1 1a   48
19d8 : b0 01 ca 8e 22 1a a2 00   0b
19e0 : 20 c2 3c 68 c9 23 f0 8c   10
19e8 : c9 25 f0 59 b0 35 20 80   e1
19f0 : 03 9d 1d 1b c9 30 90 14   f5
19f8 : c9 3a 90 08 c9 41 90 0c   04
1a00 : c9 47 b0 08 e8 e0 05 d0   e5
1a08 : e5 4c 49 14 20 c7 1a b0   f2
1a10 : 03 4c 76 80 a9 1d 85 24   c9
1a18 : a9 1b 85 25 8a 4c 79 80   47
1a20 : 00 00 00 20 80 03 20 ef   a5
1a28 : 77 20 ee 1a 20 d2 b8 8d   45
1a30 : 1d 1b 8e 1e 1b a5 16 20   ba
1a38 : d2 b8 8d 1f 1b 8e 20 1b   8a
1a40 : a2 04 4c 9a 19 8e 9b 1a   f3
1a48 : 8e 9c 1a 20 80 03 e8 e0   34
1a50 : 12 f0 b6 c9 30 90 10 c9   1c
1a58 : 32 b0 0c 29 01 6a 2e 9b   5e
1a60 : 1a 2e 9c 1a 4c 4b 1a 20   c4
1a68 : c7 1a b0 26 20 6e 86 20   fd
1a70 : b1 1a a5 24 85 3d a5 25   3f
1a78 : 85 3e 20 81 14 20 c2 3c   1a
1a80 : a2 ff 8e 22 1a ee 87 03   47
1a88 : e8 20 45 1a ce 87 03 4c   e3
1a90 : bc 1a ac 9b 1a ad 9c 1a   ae
1a98 : 4c c9 84 00 00 20 e0 87   7e
1aa0 : a0 7f 8c 00 ff a0 02 b1   93
1aa8 : 66 99 fb 00 88 10 f8 a8   18
1ab0 : 60 a5 3d 8d 20 1a a5 3e   ca
1ab8 : 8d 21 1a 60 ad 20 1a 85   b8
1ac0 : 3d ad 21 1a 85 3e 60 20   6c
1ac8 : 86 03 90 12 c9 27 b0 04   d7
1ad0 : c9 22 b0 0a c9 41 90 04   09
1ad8 : c9 5b 90 02 38 60 2c 22   2f
1ae0 : 1a 30 f9 20 bc 1a 20 ef   92
1ae8 : 77 20 dd 77 18 60 38 b0   9c
1af0 : 01 18 68 aa 68 a8 a5 17   fd
1af8 : 48 a5 16 48 98 48 8a 48   28
1b00 : b0 01 60 20 da 77 4c 15   12
1b08 : 88 68 18 69 01 85 24 68   95
1b10 : 69 00 85 25 68 85 16 68   5b
1b18 : 85 17 6c 24 00 00 00 00   c9
1b20 : 00 00 00 00 00 00 00 00   21
1b28 : 00 00 00 00 00 00 00 00   29
1b30 : 00 00 00 00 00 00 00 00   31
1b38 : 00 00 00 00 00 00 00 00   39
1b40 : 00 00 00 00 00 00 00 00   41
1b48 : 00 00 00 00 00 00 00 00   49
1b50 : 00 00 00 00 00 00 00 00   51
1b58 : 00 00 00 00 00 00 00 00   59
1b60 : 00 00 00 00 00 00 00 00   61
1b68 : 00 00 00 00 00 00 00 00   69
1b70 : 00 00 00 00 00 00 00 00   71
1b78 : 00 00 00 00 00 00 00 00   79
1b80 : 00 00 00 00 00 00 00 00   81
1b88 : 00 00 00 00 00 00 00 00   89
1b90 : 00 00 00 00 00 00 00 00   91
1b98 : 00 00 00 00 00 00 00 00   99
1ba0 : 00 00 00 00 00 00 00 00   a1
1ba8 : 00 00 00 00 00 00 00 00   a9
1bb0 : 00 00 00 00 00 00 00 00   b1
1bb8 : 00 00 00 00 00 00 00 00   b9
1bc0 : 00 00 00 00 00 00 00 00   c1
1bc8 : 00 00 00 00 00 00 00 00   c9
1bd0 : 00 00 00 00 00 00 00 00   d1
1bd8 : 00 00 00 00 00 00 00 00   d9
1be0 : 00 00 00 00 00 00 00 00   e1
1be8 : 00 00 00 00 00 00 00 00   e9
1bf0 : 00 00 00 00 00 00 00 00   f1
1bf8 : 00 00 00 00 00 00 00 00   f9
1c00 : 00 00 00 00 00 00 00 00   01
1c08 : 00 00 00 00 00 00 00 00   09
1c10 : 00 00 00 00 00 00 00 00   11
1c18 : 00 00 00 00 00 00 00 00   19
1c20 : 00 00 00 00 00 00 00 00   21
1c28 : 00 00 00 00 00 00 00 00   29
1c30 : 00 00 00 00 00 00 00 00   31
1c38 : 00 00 00 00 00 00 00 00   39
1c40 : 00 00 00 00 00 00 00 00   41
1c48 : 00 00 00 00 00 00 00 00   49
1c50 : 00 00 00 00 00 00 00 00   51
1c58 : 00 00 00 00 00 00 00 00   59
1c60 : 00 00 00 00 00 00 00 00   61
1c68 : 00 00 00 00 00 00 00 00   69
1c70 : 00 00 00 00 00 00 00 00   71
1c78 : 00 00 00 00 00 00 00 00   79
1c80 : 00 00 00 00 00 00 00 00   81
1c88 : 00 00 00 00 00 00 00 00   89
1c90 : 00 00 00 00 00 00 00 00   91
1c98 : 00 00 00 00 00 00 00 00   99
1ca0 : 00 00 00 00 00 00 00 00   a1
1ca8 : 00 00 00 00 00 00 00 00   a9
1cb0 : 00 00 00 00 00 00 00 00   b1
1cb8 : 00 00 00 00 00 00 00 00   b9
1cc0 : 00 00 00 00 00 00 00 00   c1
1cc8 : 00 00 00 00 00 00 00 00   c9
1cd0 : 00 00 00 00 00 00 00 00   d1
1cd8 : 00 00 00 00 00 00 00 00   d9
1ce0 : 00 00 00 00 00 00 00 00   e1
1ce8 : 00 00 00 00 00 00 00 00   e9
1cf0 : 00 00 00 00 00 00 00 00   f1
1cf8 : 00 00 00 00 00 00 00 00   f9
1d00 : 00 00 00 00 00 00 00 00   01
1d08 : 00 00 00 00 00 00 00 00   09
1d10 : 00 00 00 00 00 00 00 00   11
1d18 : 00 00 00 00 00 00 00 00   19
1d20 : 00 00 00 00 00 00 00 00   21
1d28 : 00 00 00 00 00 00 00 00   29
1d30 : 00 00 00 00 00 00 00 00   31
1d38 : 00 00 00 00 00 00 00 00   39
1d40 : 00 00 00 00 00 00 00 00   41
1d48 : 00 00 00 00 00 00 00 00   49
1d50 : 00 00 00 00 00 00 00 00   51
1d58 : 00 00 00 00 00 00 00 00   59
1d60 : 00 00 00 00 00 00 00 00   61
1d68 : 00 00 00 00 00 00 00 00   69
1d70 : 00 00 00 00 00 00 00 00   71
1d78 : 00 00 00 00 00 00 00 00   79
1d80 : 00 00 00 00 00 00 00 00   81
1d88 : 00 00 00 00 00 00 00 00   89
1d90 : 00 00 00 00 00 00 00 00   91
1d98 : 00 00 00 00 00 00 00 00   99
1da0 : 00 00 00 00 00 00 00 00   a1
1da8 : 00 00 00 00 00 00 00 00   a9
1db0 : 00 00 00 00 00 00 00 00   b1
1db8 : 00 00 00 00 00 00 00 00   b9
1dc0 : 00 00 00 00 00 00 00 00   c1
1dc8 : 00 00 00 00 00 00 00 00   c9
1dd0 : 00 00 00 00 00 00 00 00   d1
1dd8 : 00 00 00 00 00 00 00 00   d9
1de0 : 00 00 00 00 00 00 00 00   e1
1de8 : 00 00 00 00 00 00 00 00   e9
1df0 : 00 00 00 00 00 00 00 00   f1
1df8 : 00 00 00 00 00 00 00 00   f9
1e00 : 00 00 00 00 00 00 00 00   01
1e08 : 00 00 00 00 00 00 00 00   09
1e10 : 00 00 00 00 00 00 00 00   11
1e18 : 00 00 00 00 00 00 00 00   19
1e20 : 00 00 00 00 00 00 00 00   21
1e28 : 00 00 00 00 00 20 03 88   47
1e30 : 86 fc 20 3d 1e a5 fc 4c   80
1e38 : a1 24 4c 28 7d 20 40 1e   1a
1e40 : a2 12 a5 17 20 a3 24 e8   b9
1e48 : a5 16 4c cc cd 20 ee 1a   73
1e50 : 98 4c 75 1e 20 ee 1a 4c   aa
1e58 : 75 1e 20 ee 1a aa d0 da   b2
1e60 : 98 c9 25 b0 d5 aa 20 ab   c7
1e68 : 24 4c 75 1e 20 ee 1a 20   f6
1e70 : 3d 1e 20 a9 24 a8 68 85   2e
1e78 : 16 68 85 17 4c d4 84 20   c4
1e80 : 03 88 a5 16 85 fc a5 17   f8
1e88 : 85 fd a9 02 85 83 8a 48   e6
1e90 : 18 69 05 8d e7 20 38 e9   84
1e98 : 0b b0 02 a9 00 8d e3 20   ed
1ea0 : a9 00 8d e4 20 8d e8 20   9c
1ea8 : a5 fc a6 fd 8d e1 20 8e   ba
1eb0 : e2 20 8d e5 20 8e e6 20   15
1eb8 : 20 b4 1f 68 8d e3 20 8d   9b
1ec0 : e7 20 a9 00 8d e4 20 8d   bd
1ec8 : e8 20 a5 fc a6 fd 18 69   57
1ed0 : 0a 90 01 e8 8d e5 20 8e   25
1ed8 : e6 20 38 e9 15 b0 06 ca   9e
1ee0 : 10 03 a2 00 8a 8d e1 20   f7
1ee8 : 8e e2 20 4c b4 1f a9 08   74
1ef0 : 8d 2f 0a a2 14 20 a3 24   05
1ef8 : a9 00 8d 2e 0a 8d 22 35   ca
1f00 : a2 0c 4c cc cd 20 ee 1e   2b
1f08 : a2 19 20 ab 24 09 80 29   93
1f10 : bf 8e f0 20 4c a3 24 20   09
1f18 : 05 1f a2 18 20 ab 24 29   9b
1f20 : 7f 20 a3 24 a2 13 a9 00   86
1f28 : 20 a3 24 a0 40 a2 12 98   ca
1f30 : 20 a3 24 a9 00 20 a1 24   30
1f38 : ca 20 a3 24 88 10 ee 60   05
1f40 : a2 00 8e f0 20 a2 19 20   60
1f48 : ab 24 09 40 29 7f 20 a3   a6
1f50 : 24 20 0c ce 20 de 6a 60   c5
1f58 : 20 f4 87 e0 25 b0 0d 8e   19
1f60 : e8 03 20 09 88 8a ae e8   5c
1f68 : 03 4c a3 24 4c 28 7d a9   4e
1f70 : 01 8d ef 20 20 aa 21 f0   f5
1f78 : 08 20 86 03 d0 03 4c a3   30
1f80 : 20 20 86 03 c9 2c f0 05   7e
1f88 : c9 a4 f0 01 60 48 20 80   c9
1f90 : 03 a2 04 20 70 9e 8e e9   f3
1f98 : 03 68 10 06 20 b1 1f 4c   39
1fa0 : 81 1f 20 f2 9d 20 dc 21   a8
1fa8 : ae e9 03 20 a3 20 4c 81   7f
1fb0 : 1f 20 dc 21 a9 00 8d dd   c7
1fb8 : 20 8d de 20 8d df 20 ad   0e
1fc0 : e5 20 ac e6 20 38 ed e1   fc
1fc8 : 20 aa 98 ed e2 20 a8 08   03
1fd0 : 8a 28 20 9c 9d 8d d3 20   e0
1fd8 : 8c d4 20 10 03 ee dd 20   38
1fe0 : ad e7 20 ac e8 20 38 ed   6b
1fe8 : e3 20 aa 98 ed e4 20 a8   71
1ff0 : 08 8a 28 20 9c 9d 8d d5   e4
1ff8 : 20 8c d6 20 10 03 ee de   ab
2000 : 20 cc d4 20 90 0c d0 05   76
2008 : cd d3 20 90 05 a9 02 8d   9a
2010 : df 20 ad df 20 49 02 8d   d6
2018 : e0 20 aa bd d3 20 bc d4   45
2020 : 20 0a 8d d7 20 98 2a 8d   2e
2028 : d8 20 a8 ad d7 20 ae df   e9
2030 : 20 38 fd d3 20 8d db 20   84
2038 : 98 fd d4 20 8d dc 20 a8   9a
2040 : ad db 20 38 fd d3 20 8d   04
2048 : d9 20 98 fd d4 20 8d da   51
2050 : 20 4c 76 20 20 6c 21 ad   7d
2058 : dc 20 30 3e 20 8b 21 ad   56
2060 : d9 20 ac da 20 18 6d db   00
2068 : 20 8d db 20 98 6d dc 20   f2
2070 : 8d dc 20 20 a3 20 ae df   2d
2078 : 20 bd e1 20 dd e5 20 d0   22
2080 : d3 bd e2 20 dd e6 20 d0   26
2088 : cb 20 f2 9d ad ef 20 d0   50
2090 : 01 60 a9 00 8d ef 20 4c   9d
2098 : a3 20 ad d7 20 ac d8 20   bd
20a0 : 4c 65 20 20 f9 20 ad 6b   d9
20a8 : 11 0d 6c 11 f0 23 20 76   13
20b0 : 21 20 f9 20 4c 7f 21 ad   04
20b8 : e4 20 d0 13 a9 c7 cd e3   1b
```

```
20c0 : 20 90 0c a9 7f ed e1 20   90
20c8 : a9 02 ed e2 20 b0 02 38   4a
20d0 : 60 18 60 00 00 00 00 00   55
20d8 : 00 00 00 00 00 00 00 00   d9
20e0 : 00 00 00 00 00 00 00 00   e1
20e8 : 00 00 00 00 00 00 00 00   e9
20f0 : 00 7f bf df ef f7 fb fd   46
20f8 : fe 20 b7 20 b0 d3 ac e3   1c
2100 : 20 ae e2 20 ad e1 20 48   2f
2108 : 29 07 8d ed 20 68 4a 4a   d9
2110 : 4a 18 7d e8 21 48 98 29   ec
2118 : 0f aa 68 18 7d eb 21 8d   70
2120 : ec 20 b9 fb 21 69 00 8d   83
2128 : eb 20 20 5d 21 20 a9 24   d9
2130 : ae ed 20 a4 83 f0 1b c0   1f
2138 : 02 f0 0b 90 0f 3d 1c 9d   0e
2140 : a8 f0 02 a0 01 60 5d 1c   b6
2148 : 9d 4c 55 21 1d 1c 9d 4c   47
2150 : 55 21 3d f1 20 48 20 5d   43
2158 : 21 68 4c a1 24 a2 12 ad   f0
2160 : eb 20 20 a3 24 e8 ad ec   f2
2168 : 20 4c a3 24 ad df 20 d0   18
2170 : 1f ae dd 20 d0 09 ee e1   37
2178 : 20 d0 03 ee e2 20 60 ad   ab
2180 : e1 20 d0 03 ce e2 20 ce   28
2188 : e1 20 60 ac df 20 d0 e1   2d
2190 : ac de 20 d0 09 ee e3 20   a5
2198 : d0 03 ee e4 20 60 ad e3   c5
21a0 : 20 d0 03 ce e4 20 ce e3   15
21a8 : 20 60 ae f0 20 d0 03 4c   ef
21b0 : 79 a0 a2 01 86 83 8d ee   db
21b8 : 20 20 86 03 f0 13 c9 2c   11
21c0 : f0 0f ad ee 20 f0 07 20   67
21c8 : 86 03 c9 a4 f0 0d 20 f4   b9
21d0 : 87 e0 03 90 03 4c 28 7d   c8
21d8 : 86 83 e8 60 a2 07 bd 31   22
21e0 : 11 9d e1 20 ca 10 f7 60   0a
21e8 : 00 20 40 00 50 a0 f0 40   57
21f0 : 90 e0 30 80 d0 20 70 c0   5e
21f8 : 10 60 b0 00 00 00 00 01   67
2200 : 01 01 02 02 02 03 03 03   8d
2208 : 04 04 04 05 05 05 05 06   49
2210 : 06 06 07 07 07 08 08 08   9d
2218 : 09 09 09 0a 0a 0a 0a 0b   59
2220 : 0b 0b 0c 0c 0c 0d 0d 0d   ad
2228 : 0e 0e 0e 0f 0f 0f 0f 10   69
2230 : 10 10 11 11 11 12 12 12   bd
2238 : 13 13 13 14 14 14 14 15   79
2240 : 15 15 16 16 16 17 17 17   cd
2248 : 18 18 18 19 19 19 19 1a   89
2250 : 1a 1a 1b 1b 1b 1c 1c 1c   dd
2258 : 1d 1d 1d 1e 1e 1e 1e 1f   99
2260 : 1f 1f 20 20 20 21 21 21   ed
2268 : 22 22 22 23 23 23 23 24   a8
2270 : 24 24 25 25 25 26 26 26   fd
2278 : 27 27 27 28 28 28 28 29   b8
2280 : 29 29 2a 2a 2a 2b 2b 2b   0d
2288 : 2c 2c 2c 2d 2d 2d 2d 2e   c8
2290 : 2e 2e 2f 2f 2f 30 30 30   1d
2298 : 31 31 31 32 32 32 32 33   d8
22a0 : 33 33 34 34 34 35 35 35   2d
22a8 : 36 36 36 37 37 37 37 38   e8
22b0 : 38 38 39 39 39 3a 3a 3a   3d
22b8 : 3b 3b 3b 3c 3c 3c 3c 3d   f8
22c0 : 3d 3d 3e a9 00 20 aa 21   4f
22c8 : a2 1f 20 52 9e 20 06 9e   8c
22d0 : 8c 54 11 8d 55 11 20 06   e7
22d8 : 9e 8c 56 11 b0 09 ad 54   27
22e0 : 11 8d 56 11 ad 55 11 8d   54
22e8 : 57 11 20 06 9e 8c 5c 11   73
22f0 : 8d 5d 11 20 06 9e 8c 5e   b8
22f8 : 11 8d 5f 11 20 06 9e 85   82
2300 : 77 98 a4 77 20 77 9a a2   49
2308 : 2d a0 2b 20 7c 9d 90 06   57
2310 : 20 1c 23 4c 4b 23 20 2a   33
2318 : 23 4c 4b 23 a9 68 a0 01   fb
2320 : 20 70 9d 9d 31 11 98 9d   cd
2328 : 32 11 a2 03 bd 54 11 9d   ea
2330 : 58 11 ca 10 f7 a9 90 20   15
2338 : f3 9a a2 07 bd 54 11 9d   00
2340 : 60 11 ca 10 f7 20 50 67   6e
2348 : 4c f2 9d a2 02 20 1e 9e   a0
2350 : 8a d0 03 4c 28 7d 8e 20   76
2358 : 12 18 ad 20 12 6d 5c 11   06
2360 : 8d 5c 11 90 03 ee 5d 11   b1
2368 : a2 2d a0 2b 20 7c 9d b0   ec
2370 : 09 20 50 67 20 b1 1f 4c   2f
2378 : 59 23 a0 2d 20 52 67 4c   fb
2380 : b1 1f a9 00 20 aa 21 a2   4c
2388 : 1f 20 6d 9e a2 2b 20 52   8f
2390 : 9e 20 06 9e 8c 54 11 8d   5e
2398 : 55 11 20 1c 9e e0 02 90   1c
23a0 : 03 4c 28 7d 8e 6c 11 8a   29
23a8 : 48 20 89 63 68 d0 1c f0   2e
23b0 : 03 20 0b 64 20 b1 1f ad   7a
23b8 : 4e 11 d0 f5 a2 04 bd 5b   7a
23c0 : 11 9d 30 11 ca d0 f7 8e   fe
23c8 : 6c 11 60 a6 83 e0 02 90   12
23d0 : 08 a2 00 8e 6c 11 4c 28   cc
23d8 : 7d a2 00 ad 49 11 4a 90   c4
23e0 : 02 a2 02 bd 60 11 8d 5a   e5
23e8 : 11 bd 61 11 8d 5b 11 a9   9e
23f0 : 00 a2 03 9d 56 11 ca 10   ef
23f8 : fa a2 07 bd 31 11 48 ca   0f
2400 : 10 f9 20 b1 1f a2 00 68   23
2408 : 9d 31 11 e8 e0 08 d0 f7   21
2410 : ad 5a 11 d0 05 ce 5b 11   9f
2418 : 30 a2 ce 5a 11 a2 25 a0   94
2420 : 1b ad 49 11 4a 90 02 a0   f9
2428 : 19 a9 00 4a 48 20 6d 9d   d6
2430 : 9d 31 11 98 9d 32 11 68   3e
2438 : 90 02 09 a0 e8 e8 a0 19   aa
2440 : 4e 49 11 90 02 a0 1b 2e   77
2448 : 49 11 e0 27 f0 dd a2 06   cb
2450 : 0a f0 bd 90 08 fe 31 11   b3
2458 : d0 03 fe 32 11 0a ca ca   d2
2460 : 10 f1 30 95 a2 1a 20 ab   fa
2468 : 24 29 0f 85 fc 20 91 24   f5
2470 : 86 fd 20 5c 79 20 91 24   b0
2478 : 18 8a 69 68 a2 23 20 a3   48
2480 : 24 a6 fd bd 4c 6a 0a 0a   83
2488 : 0a 0a 05 fc a2 1a 4c a3   ec
2490 : 24 20 f4 87 e0 11 b0 06   58
2498 : e0 00 f0 02 ca 60 4c 28   26
24a0 : 7d a2 1f 20 45 a8 4c cc   9f
24a8 : cd a2 1f 20 45 a8 4c da   13
24b0 : cd ae f0 20 f0 03 4c 58   1d
24b8 : 69 4c 79 a0 20 f1 1a 20   f4
24c0 : f7 87 e0 02 90 10 f0 03   47
24c8 : 4c 28 7d 20 57 25 90 02   71
24d0 : a0 00 a9 00 f0 0a 8a 0a   78
24d8 : aa bd e1 20 a8 bd e2 20   21
24e0 : aa 20 09 1b 8a 18 4c c9   6e
24e8 : 84 20 2d 25 2c 00 d6 10   ab
24f0 : fb ad 01 d6 a2 3f 8e 00   3b
24f8 : ff 91 fc a2 00 8e 00 ff   c8
2500 : c8 d0 e9 e6 fd a5 fd c5   18
2508 : fe d0 e1 60 20 2d 25 a2   38
2510 : 3f 8e 00 ff b1 fc 20 ac   73
2518 : 15 2c 00 d6 10 fb 8d 01   37
2520 : d6 c8 d0 eb e6 fd a5 fd   fd
2528 : c5 fe d0 e3 60 20 86 03   44
2530 : f0 06 20 12 88 4c 3c 25   94
2538 : a9 bf a0 00 85 fd 84 fc   3d
2540 : 18 69 40 85 fe a9 00 a2   50
2548 : 12 20 a3 24 e8 20 cc cd   36
2550 : a8 a2 1f 8e 00 d6 60 a2   60
2558 : 03 86 83 20 b7 20 b0 04   ca
2560 : 20 a3 20 18 60 08 20 51   c6
2568 : 42 a9 03 8d 00 0a a9 40   69
2570 : 8d 01 0a 78 20 56 e0 58   f8
2578 : 28 f0 10 20 80 03 a9 2b   3e
2580 : 8d 19 03 a9 15 8d 18 03   b4
2588 : 4c 30 30 20 81 14 ad 06   78
2590 : d5 29 f0 09 04 8d 06 d5   c8
2598 : a9 60 8d 00 13 20 a3 25   e0
25a0 : 4c ca 40 20 64 3b a9 e2   f2
25a8 : 20 af 02 88 a2 7f a9 24   47
25b0 : 4c af 02 20 81 14 ad 0e   e4
25b8 : dc 09 80 8d 0e dc ad 0f   87
25c0 : dc 29 7f 8d 0f dc 60 20   5c
25c8 : b3 25 a2 03 bd 08 dc 9d   e2
25d0 : 30 26 ca 10 f7 a0 03 a2   9e
25d8 : 00 b9 30 26 48 29 f0 4a   ac
25e0 : 4a 4a 4a 9d 34 26 e8 68   7f
25e8 : 29 0f 9d 34 26 e8 88 10   73
25f0 : e8 2c 33 26 10 13 ad 34   39
25f8 : 26 29 01 18 69 01 8d 34   33
2600 : 26 ad 35 26 69 02 8d 35   56
2608 : 26 ad 35 26 c9 0a 90 06   52
2610 : 38 e9 0a ee 34 26 8d 35   b2
2618 : 26 ad 34 26 c9 02 d0 0f   f5
2620 : ad 35 26 c9 04 d0 08 a9   65
2628 : 00 8d 34 26 8d 35 26 60   9d
2630 : 00 00 00 00 00 00 00 00   31
2638 : 00 00 00 00 00 20 c7 25   a3
2640 : 20 52 14 a6 fb e0 07 f0   28
2648 : 03 4c 28 7d 20 ac 15 a0   28
2650 : 02 a9 00 20 9c 26 0a 0a   62
2658 : 0a 0a 20 9c 26 99 31 26   43
2660 : 88 10 ee ad 33 26 c9 20   2d
2668 : 90 02 e9 06 c9 10 90 05   9e
2670 : e9 06 8d 33 26 c9 18 b0   99
2678 : 20 c9 0c 90 05 69 73 8d   16
2680 : 33 26 a2 03 bd 30 26 9d   01
2688 : 08 dc ca d0 f7 8a a2 06   35
2690 : 20 9c 26 8d 08 dc 4c c7   62
2698 : 25 4c 28 7d 85 fb bd 2d   27
26a0 : 1c e8 20 90 03 b0 a2 e9   5f
26a8 : 2f 05 fb 60 20 c7 25 a2   7f
26b0 : 00 a0 00 bd 34 26 18 69   60
26b8 : 30 99 2d 1b e8 c8 bd 34   98
26c0 : 26 18 69 30 99 2d 1b a9   15
26c8 : 3a c8 99 2d 1b c8 e8 e0   d0
26d0 : 06 d0 e0 a9 2e 99 2c 1b   42
26d8 : e8 bd 34 26 18 69 30 99   31
26e0 : 2d 1b a9 0a 85 fd 4c 56   6d
26e8 : 2d a9 ff a0 00 a6 d1 f0   5c
26f0 : 09 18 79 00 10 c8 c5 d2   68
26f8 : 90 f7 4c d4 84 20 ef 77   29
2700 : 24 0f 10 06 20 7c 18 20   f7
2708 : ef 77 20 da 77 20 15 88   f4
2710 : 4c da 6b a0 ff 20 1e 2d   8c
2718 : a9 2d 8d 7c 14 a9 1b 8d   61
2720 : 7d 14 20 5f 14 a6 fb f0   e4
2728 : 27 a9 00 9d 2d 1b a5 3d   94
2730 : 8d 53 27 a5 3e 8d 54 27   d5
2738 : a9 2d 85 3d a9 1b 85 3e   87
2740 : 20 0a 43 20 ef 77 ad 53   52
2748 : 27 85 3d ad 54 27 85 3e   48
2750 : 4c 57 19 00 00 a5 00 09   cd
2758 : 40 85 00 a5 01 29 bf 85   73
2760 : 01 60 a5 00 29 bf 85 00   a1
2768 : 60 a9 ff d0 02 a9 00 8d   3f
2770 : b2 27 a2 b3 a0 27 20 66   65
2778 : af 20 86 03 f0 2d c9 29   2b
2780 : f0 29 20 5c 79 c9 2c f0   11
2788 : 06 20 d7 77 4c 94 27 a0   ca
2790 : 00 20 d4 84 a9 b3 a0 27   6f
2798 : 20 54 af 08 2c b2 27 10   e4
27a0 : 05 28 b0 d5 90 cc 28 90   d1
```

Listing 1. (Fortsetzung)

```
27a8 : d0 b0 c7 a9 b3 a0 27 4c  6d
27b0 : 63 af 00 00 00 00 00 00  eb
27b8 : 20 ee 1a 98 29 01 49 01  ab
27c0 : a8 88 20 09 1b 98 18 4c  45
27c8 : 3c 79 20 ee 1a aa d0 04  e9
27d0 : c0 0a 90 03 4c 28 7d 8c  2f
27d8 : 1a 28 8c 1b 28 20 5c 79  75
27e0 : 20 d7 77 20 09 1b ae 1a  26
27e8 : 28 f0 08 20 17 8b ce 1a  cc
27f0 : 28 d0 f8 20 51 af 08 a9  c8
27f8 : 76 a0 8f 20 5d af 20 1b  b0
2800 : af 20 2d af ae 1b 28 f0  47
2808 : 08 20 38 8b ce 1b 28 d0  a8
2810 : f8 28 90 03 20 33 af 4c  94
2818 : 4b af 00 00 20 ee 1a 20  5d
2820 : 09 88 86 fd 20 3d 1e a0  74
2828 : 00 a2 1f c4 fd d0 10 f0  62
2830 : 03 20 f1 1a a9 2d 85 16  49
2838 : a9 1c 85 17 4c 61 2d 20  f8
2840 : da cd 99 2d 1c c8 d0 e3  20
2848 : 20 55 14 a4 fb 84 fd 88  99
2850 : b9 2d 1c 20 ca 2c 99 2d  7a
2858 : 1c 98 d0 f3 f0 d3 20 55  4c
2860 : 14 a4 fb 84 fd 88 b9 2d  bb
2868 : 1c 20 74 28 99 2d 1c 98  5b
2870 : d0 f3 f0 bd c9 5e f0 1e  bd
2878 : c9 20 b0 03 69 40 60 c9  8c
2880 : 40 90 fb c9 60 b0 03 69  ab
2888 : 80 60 c9 80 90 ee c9 c0  e4
2890 : 90 f5 18 69 c0 60 a9 ff  04
2898 : 60 00 a9 ff d0 02 a9 00  26
28a0 : 8d 99 28 20 12 88 20 4f  8c
28a8 : 14 20 3d 1e a0 00 a2 1f  b2
28b0 : c4 fb d0 01 60 b9 2d 1c  87
28b8 : 2c 99 28 10 03 20 ca 2c  72
28c0 : 20 cc cd c8 d0 ea 20 55  62
28c8 : 14 a4 fb 98 f0 0b 88 b9  3d
28d0 : 2d 1c 29 7f c9 20 f0 f3  8f
28d8 : c8 84 fd 4c 31 28 20 55  6b
28e0 : 14 a4 fb f0 f4 88 4c d9  dc
28e8 : 28 20 55 14 a4 fb f0 e9  ba
28f0 : a0 00 a2 01 20 0f 29 4c  11
28f8 : d9 28 20 55 14 a2 ff a0  30
2900 : 00 e8 e4 fb f0 d3 bd 2d  2c
2908 : 1c 29 7f c9 20 f0 f2 e4  f1
2910 : fb f0 c6 bd 2d 1c 99 2d  61
2918 : 1c e8 c8 d0 f2 f0 ba 20  d6
2920 : ee 1a ae d5 03 a0 00 a9  0a
2928 : 16 20 74 ff 48 ae d5 03  c3
2930 : c8 a9 16 20 74 ff aa 68  19
2938 : a8 4c e1 24 20 12 88 84  c1
2940 : fa 85 fb 20 0f 88 a2 fa  b5
2948 : 8e b9 02 48 98 a0 00 78  bc
2950 : ae d5 03 20 77 ff c8 68  19
2958 : ae d5 03 20 77 ff 58 60  4f
2960 : 4c 28 7d d0 03 4c 24 ca  f2
2968 : 20 f4 87 e0 19 b0 f1 86  ec
2970 : fc 20 86 03 f0 0e 20 09  90
2978 : 88 e0 19 b0 e3 e4 fc 90  47
2980 : df 8a d0 02 a6 fc a5 fc  fc
2988 : 85 e5 86 e4 20 50 c1 a9  1d
2990 : 00 4c 0c 56 20 a6 29 a5  ab
2998 : d0 d0 0b a5 d3 29 ef a8  df
29a0 : d0 04 a5 d1 f0 f1 a9 00  5b
29a8 : 85 d0 85 d1 60 f0 08 20  1f
29b0 : f4 87 e0 0a 4c e7 60 a2  ac
29b8 : ff a0 00 e8 a9 08 85 77  04
29c0 : bd 00 10 f0 19 86 fc 85  64
29c8 : 78 a2 06 bd e3 29 ca d0  1f
29d0 : 02 05 fc 20 69 92 8a 10  0d
29d8 : f2 20 3f 61 a6 fc e0 09  be
29e0 : d0 d9 60 2c 30 20 59 45  2e
29e8 : 4b 58 20 86 03 a8 f0 59  24
29f0 : a2 00 86 fb 8e 23 30 c9  0a
29f8 : 22 d0 11 e6 fb 20 9c 5d  91
2a00 : c9 22 d0 08 a4 fb f0 04  05
2a08 : 20 80 03 98 9d 2d 1b e8  be
2a10 : a8 d0 ea a5 2e a6 2d c8  ee
2a18 : d0 23 ad 23 30 f0 01 60  99
2a20 : 20 98 55 a0 02 20 ec 42  4f
2a28 : aa c8 20 ec 42 20 25 51  38
2a30 : 20 b5 4b a0 00 20 ec 42  4b
2a38 : aa c8 20 ec 42 85 62 86  d3
2a40 : 61 20 ec 42 d0 15 8d 23  67
2a48 : 30 60 c8 e8 bd 2d 1b f0  8b
2a50 : c9 20 ec 42 dd 2d 1b f0  42
2a58 : f1 d0 29 88 84 fa a0 04  b7
2a60 : a2 00 20 ec 42 f0 cc c9  1a
2a68 : 22 d0 06 a9 01 45 fa 85  da
2a70 : fa a5 fb c5 fa d0 0d 88  70
2a78 : 20 ec 42 c8 c9 fe f0 04  18
2a80 : c9 ce d0 cd c8 d0 d9 60  d9
2a88 : a9 00 f0 0b 85 2e a9 00  3f
2a90 : 85 2d a9 00 8d 89 2a 60  a5
2a98 : ad 89 2a d0 0a a5 2d 8d  4c
2aa0 : 8f 2a a5 2e 8d 89 2a 60  02
2aa8 : 20 98 2a 20 d5 2a 85 2d  c2
2ab0 : 84 2e 4c 2c 91 f0 18 20  25
2ab8 : 12 88 e6 16 d0 02 e6 17  72
2ac0 : 20 64 50 20 98 2a a5 61  5f
2ac8 : 85 2d a5 62 85 2e 60 20  25
2ad0 : d5 2a 4c c3 2a ad 10 12  ba
2ad8 : ac 11 12 38 e9 02 b0 01  0c
2ae0 : 88 85 61 84 62 60 f0 16  2d
2ae8 : c9 f9 f0 01 60 20 45 a8  78
2af0 : a2 4c bd a8 ce 9d 00 10  37
2af8 : ca 10 f7 4c 80 03 a0 01  f6
2b00 : 98 91 2d 20 4f 4f 4c 82  56
2b08 : 4f 20 45 a8 4c 6f cd 20  85
2b10 : 45 a8 4c 9f cd d8 20 3d  4f
2b18 : f6 c9 ff d0 10 20 a1 3a  0a
2b20 : a9 40 8d 18 03 a9 fa 8d  d4
2b28 : 19 03 20 36 30 4c 40 fa  ee
2b30 : a9 00 d0 02 a9 ff 8d 1f  5d
2b38 : 17 20 fc 8b 20 5c 79 20  1b
2b40 : d7 77 2c 1f 17 30 14 20  45
2b48 : f9 8b 20 5b 2b a9 5e a0  35
2b50 : 00 20 24 8a a9 59 a0 00  a3
2b58 : 4c 2e 88 a9 59 a0 00 20  ed
2b60 : 1e 8a 4c fb 8c 20 fc 8b  2b
2b68 : 20 fb 8c a9 59 a0 00 4c  11
2b70 : 2e 88 20 aa 21 20 0f 88  a0
2b78 : 8c b5 2c 8d b6 2c 98 29  1d
2b80 : 07 8d b8 2c 20 09 88 8e  8b
2b88 : b7 2c 20 4f 14 a5 fb f0  88
2b90 : 2a a2 00 f0 27 ae b9 2c  51
2b98 : ad b5 2c 18 69 08 8d b5  a6
2ba0 : 2c 90 03 ee b6 2c a9 02  2a
2ba8 : cd b6 2c 90 0e d0 07 a9  c4
2bb0 : 78 cd b5 2c 90 05 e8 e4  a0
2bb8 : fb 90 01 60 8e b9 2c bd  2b
2bc0 : 2d 1c 20 ca 2c a2 00 86  42
2bc8 : fe 0a 26 fe 0a 26 fe 0a  17
2bd0 : 26 fe 85 fd c6 83 a2 01  ab
2bd8 : 8e 00 ff 24 f1 30 0c ad  17
2be0 : 2c 0a 29 02 a8 d0 04 a9  10
2be8 : d0 d0 02 a9 d8 18 65 fe  b8
2bf0 : 85 fe a0 00 a2 00 b1 fd  09
2bf8 : 24 83 10 02 49 ff 9d ba  a3
2c00 : 2c e8 a9 00 9d ba 2c e8  3d
2c08 : c8 c0 08 d0 e9 ca 20 81  c5
2c10 : 14 ac b8 2c 98 f0 0d 5e  30
2c18 : b9 2c 7e ba 2c 88 d0 f7  18
2c20 : ca ca 10 ed ac b7 2c ae  a8
2c28 : b6 2c ad b5 2c 4a 4a 4a  e9
2c30 : c6 83 18 7d e8 21 48 98  57
2c38 : 29 0f aa 68 18 7d eb 21  00
2c40 : 8d ec 20 b9 fb 21 69 00  f1
2c48 : 8d eb 20 a0 00 20 5d 21  a0
2c50 : 20 d8 cd 24 83 10 06 ae  02
2c58 : b8 2c 3d a5 2c 59 ba 2c  fb
2c60 : 99 ba 2c 20 d8 cd 24 83  f9
2c68 : 10 06 ae b8 2c 3d ad 2c  fa
2c70 : c8 59 ba 2c 99 ba 2c 20  79
2c78 : 5d 21 88 b9 ba 2c 20 ca  e2
2c80 : cd c8 b9 ba 2c 20 cc cd  0a
2c88 : c8 c0 10 f0 11 a9 50 18  a2
2c90 : 6d ec 20 90 03 ee eb 20  25
2c98 : 8d ec 20 4c 4d 2c e6 83  06
2ca0 : e6 83 4c 95 2b 00 80 c0  44
2ca8 : e0 f0 f8 fc fe ff 7f 3f  4a
2cb0 : 1f 0f 07 03 01 00 00 00  89
2cb8 : 00 00 00 00 00 00.00 00  b9
2cc0 : 00 00 00 00 00 00 00 00  c1
2cc8 : 00 00 48 68 30 12 c9 20  e2
2cd0 : 90 0b c9 60 90 04 29 df  f2
2cd8 : b0 02 29 3f 60 09 80 60  cd
2ce0 : 29 7f c9 20 90 09 c9 7f  b7
2ce8 : d0 02 a9 5e 09 40 60 09  16
2cf0 : c0 60 20 12 88 8c 7c 14  32
2cf8 : 8d 7d 14 20 87 14 8e 77  8f
2d00 : 14 20 5c 79 20 ef 77 4c  63
2d08 : 64 14 20 55 14 a9 00 a6  05
2d10 : fb 9d 2d 1c a9 2d 85 61  86
2d18 : a9 1c 85 62 a0 00 85 fd  99
2d20 : ad 27 03 c9 14 d0 03 4c  c7
2d28 : b0 2d a9 de 8d 26 03 a9  1e
2d30 : 14 8d 27 03 a9 00 8d f5  f2
2d38 : 14 98 f0 0c 30 0a a5 fd  3c
2d40 : 20 32 8e 20 04 56 a0 04  9e
2d48 : c0 ff f0 a5 84 4b 84 11  cf
2d50 : 20 44 51 20 57 19 20 f1  8d
2d58 : 1a a9 2d 85 16 a9 1b 85  69
2d60 : 17 a2 ff 8e dd 14 a6 fd  af
2d68 : 4c a7 14 20 ee 1a 85 62  2c
2d70 : 84 61 20 09 1b a0 02 20  cd
2d78 : ec 42 aa c8 20 ec 42 4c  54
2d80 : 1e 2d 20 12 88 8d 1a 12  01
2d88 : 8c 19 12 60 d0 0b 20 c2  9d
2d90 : 3c a9 ff 8d 22 0a 4c 80  f7
2d98 : 03 c9 91 d0 04 a9 00 f0  6e
2da0 : f2 c9 fe d0 0b 20 80 03  0a
2da8 : c9 43 d0 04 a9 40 d0 e3  6f
2db0 : 4c 6c 79 a2 ff d0 02 a2  b9
2db8 : 00 8e ed 2d 20 81 14 20  bf
2dc0 : bd ff a2 04 a0 07 20 ba  de
2dc8 : ff 20 c0 ff a2 00 20 c9  46
2dd0 : ff 20 86 03 2c ed 2d 30  29
2dd8 : 06 20 e2 50 4c e2 2d 20  82
2de0 : 57 55 20 98 55 a9 00 20  e0
2de8 : c3 ff 4c cc ff 00 20 f4  c2
2df0 : 87 8e ee 03 20 09 88 8e  64
2df8 : c5 2d ad ee 03 8d c3 2d  a3
2e00 : 60 ad 10 12 ae 11 12 38  a9
2e08 : e5 2d b0 01 ca 18 69 01  e5
2e10 : 90 01 e8 a8 8a 38 e5 2e  ce
2e18 : 4c e5 24 ac d5 03 4c d4  46
2e20 : 84 20 e1 a7 d0 17 78 a9  1d
2e28 : 2e 8d 01 0a 8d a7 25 a9  9c
2e30 : 3a 8d 00 0a 8d af 25 20  9d
2e38 : 8b 25 4c 3a 2e 60 20 e1  da
2e40 : a7 d0 fa a9 00 78 85 fe  1b
2e48 : 85 24 a5 2d 85 fa a5 2e  11
2e50 : 85 fb ad 10 12 85 fc ad  dd
2e58 : 11 12 85 fd 20 9a 35 24  87
2e60 : d7 30 03 20 c4 77 58 4c  16
2e68 : d9 51 ad 73 2e 49 ff 8d  0c
2e70 : 73 2e 60 00 08 2c 73 2e  1f
2e78 : 30 04 28 4c 51 51 28 10  9e
2e80 : 1e c9 ff 10 1a 24 11 30  ec
2e88 : 16 c9 8f d0 07 a9 12 20  c7
2e90 : 0c 56 a9 8f c9 fe f0 61  3f
2e98 : c9 ce f0 70 4c 63 51 c9  cb
2ea0 : 0d f0 0f c9 14 f0 0b c9  ab
2ea8 : 8d f0 07 c9 1b f0 03 4c  86
```

```
2eb0 : 32 51 8d ed 03 a5 4b 48  c7
2eb8 : a5 11 48 a5 55 48 8e ee  5c
2ec0 : 03 8c ef 03 20 45 a8 20  75
2ec8 : 7d ff 22 12 43 48 52 24  18
2ed0 : 28 00 20 81 14 ae ed 03  a5
2ed8 : 20 32 8e 20 45 a8 20 7d  ce
2ee0 : ff 29 92 00 ae ee 03 ac  e0
2ee8 : ef 03 68 85 55 68 49 ff  e1
2ef0 : 85 11 68 85 4b a9 22 d0  f5
2ef8 : b6 aa c8 20 ec 42 d0 03  64
2f00 : 4c 32 51 84 4b c9 27 b0  4b
2f08 : 16 4c 95 51 aa c8 20 ec  1f
2f10 : 42 d0 03 4c 32 51 84 4b  5b
2f18 : c9 0b b0 07 4c af 51 a2  41
2f20 : 00 f0 02 a2 ff 8d ed 03  97
2f28 : 8e ee 03 8c ef 03 a9 12  62
2f30 : 20 0c 56 ad ed 03 ae ee  31
2f38 : 03 ac ef 03 20 a3 3b 29  4c
2f40 : 7f 20 0c 56 a9 92 20 0c  65
2f48 : 56 ad ed 03 c9 56 d0 03  e9
2f50 : 20 94 29 a9 00 4c 2e 51  f8
2f58 : a0 ff 20 1e 2d 20 ee 1a  87
2f60 : 20 09 88 8a 48 a5 16 8d  9d
2f68 : fb 31 a5 17 8d fc 31 20  0e
2f70 : 09 88 86 15 20 fd 90 20  76
2f78 : 57 19 68 85 fc a2 00 20  4c
2f80 : e5 90 9d 2d 1b e8 e4 fc  41
2f88 : f0 0a cd fb 31 d0 f0 ac  27
2f90 : fc 31 d0 eb cd fb 31 d0  f9
2f98 : 06 ac fc 31 d0 01 ca 20  da
2fa0 : 09 1b 86 fd 20 58 56 4c  4f
2fa8 : 56 2d 20 ef 77 24 0f 30  d0
2fb0 : 06 20 15 88 4c e2 59 20  9e
2fb8 : 55 14 a9 fe 8d 2d 1b a9  63
2fc0 : 73 8d 2e 1b a9 22 8d 2f  29
2fc8 : 1b a2 02 8e 23 30 a6 fb  cd
2fd0 : a9 22 9d 2d 1c a9 00 85  b1
2fd8 : fb e8 d0 05 f0 17 bd 2d  35
2fe0 : 1c 9d 30 1b ca d0 f7 bd  c9
2fe8 : 2d 1c 9d 30 1b 20 13 2a  e4
2ff0 : ad 23 30 d0 03 4c 15 5a  f1
2ff8 : 98 18 65 61 85 3d a9 00  0b
3000 : 65 62 85 3e 20 86 03 f0  e4
3008 : 06 20 84 2a 4c f0 2f a0  cf
3010 : 02 20 ec 42 85 3b c8 20  3b
3018 : ec 42 85 3c a9 1b 85 18  c8
3020 : 4c 10 5a 00 20 1d 5a 20  a0
3028 : 86 03 20 aa 2f 4c f6 4a  53
3030 : a9 40 a2 30 d0 04 a9 6e  59
3038 : a2 f6 8e 29 03 8d 28 03  61
3040 : 8d 4b 30 a9 01 08 ad 4b  52
3048 : 30 28 60 00 a4 41 a5 42  14
3050 : 4c c9 84 a4 3b a5 3c 4c  a1
3058 : c9 84 20 da 77 20 59 94  ce
3060 : a9 1b a0 95 20 89 8a 4c  83
3068 : 4c 8b 20 da 77 20 ca 89  94
3070 : a9 76 a0 30 d0 0c 7f 5e  ab
3078 : 5b d8 aa 20 da 77 a9 85  09
3080 : a0 30 4c 08 8a 86 65 2e  1b
3088 : e0 d2 20 da 77 a9 93 a0  89
3090 : 30 d0 ef 7b 0e fa 35 0f  3f
3098 : 20 cb 30 a5 2d 48 a5 2e  67
30a0 : 48 8a 91 fc b1 2d c8 51  7b
30a8 : 2d 49 cd 88 91 2d e6 2d  77
30b0 : d0 02 e6 2e ad 11 12 c5  38
30b8 : 2e d0 05 ad 10 12 c5 2d  48
30c0 : d0 e2 68 85 2e 68 85 2d  63
30c8 : 4c 81 14 20 12 88 8c aa  cb
30d0 : 30 8c f6 30 aa a9 3f 8d  1a
30d8 : 00 ff ad 10 12 85 fc ad  e2
30e0 : 11 12 85 fd a0 00 60 20  e7
30e8 : cb 30 8a a6 fc d0 02 c6  2f
30f0 : fd c6 fc 51 fc 49 cd 91  2e
30f8 : fc a6 fc e4 2d d0 ec a6  7d
3100 : fd e4 2e d0 e6 ad 10 12  55
3108 : 85 fc ad 11 12 85 fd c6  6c
3110 : fd a0 fd b1 fc f0 03 4c  0f
3118 : 21 2e c8 d0 f6 20 81 14  3b
3120 : 4c fe 2a 20 f1 1a 20 f7  da
3128 : 87 8a f0 0a ca d0 10 a4  2e
3130 : 39 a5 3a 4c 3c 31 ac 12  78
3138 : 12 ad 13 12 4c e0 24 4c  1d
3140 : 28 7d 20 f1 1a 20 f7 87  ff
3148 : 8a f0 0a ca d0 f1 a4 2f  b4
3150 : a5 30 4c 3c 31 a4 2d a5  e0
3158 : 2e 4c 3c 31 a9 ff 20 0d  17
3160 : 19 a5 fd d0 01 60 20 81  7c
3168 : 14 85 90 a9 08 20 b1 ff  e0
3170 : 24 90 10 05 a2 05 4c 3c  7d
3178 : 4d a9 6f 20 93 ff 24 90  65
3180 : 30 f2 a2 00 bd 2d 1b 20  c4
3188 : a8 ff a5 90 d0 05 e8 e4  4e
3190 : fd d0 f1 20 ae ff 4c 43  18
3198 : 92 20 f4 87 e0 08 90 9f  38
31a0 : e0 0c b0 9b 8e 6c 31 60  f8
31a8 : 20 d9 84 ad 49 3c d0 37  b4
31b0 : 20 f6 14 a6 fb a9 00 9d  6d
31b8 : 2d 1d a5 3d 8d fb 31 a5  4e
31c0 : 3e 8d fc 31 a9 2d 85 3d  bf
31c8 : a9 1d 85 3e 20 86 03 90  8c
31d0 : 2d 20 0a 43 20 86 03 20  7b
31d8 : 9f 15 20 b5 4b 20 86 03  96
31e0 : aa f0 0d c9 3a f0 03 4c  4f
31e8 : 6c 79 20 80 03 4c d7 31  7d
31f0 : ad fb 31 85 3d ad fc 31  2f
31f8 : 85 3e 60 00 00 00 20 27  83
3200 : 32 ad fb 31 38 e5 61 8d  81
3208 : fd 31 20 86 03 20 a0 50  cb
3210 : a5 16 c5 3b d0 09 a5 17  b3
3218 : c5 3c d0 03 4c 28 7d a9  df
3220 : ff 8d 49 3c 4c e5 4d a5  34
3228 : 3b 85 16 a5 3c 85 17 4c  45
3230 : 64 50 4c 28 7d 20 d9 84  1e
3238 : 20 ef 77 20 e2 18 20 09  b3
3240 : 88 8a f0 ee e0 a1 b0 ea  db
3248 : 8e 7f 32 20 5c 79 20 7f  38
3250 : 18 a5 eb 8d 7c 32 a5 ec  b1
3258 : 8d 7d 32 20 04 56 20 7f  a7
3260 : 18 ae 7f 32 8e 81 32 20  f3
3268 : 52 14 20 83 32 20 5c 79  c5
3270 : a9 ff 8d 80 32 a5 fb 8d  e8
3278 : 7e 32 d0 3f 00 00 00 00  2b
3280 : 00 00 00 20 81 14 ad 7c  ed
3288 : 32 48 ad 7d 32 4c 86 cc  33
3290 : 20 81 14 ad 7c 32 48 ad  01
3298 : 7d 32 18 6d 82 32 4c 86  da
32a0 : cc a9 a0 8d 7f 32 a9 00  4b
32a8 : 8d 80 32 8d 81 32 20 d9  91
32b0 : 84 20 86 03 c9 22 d0 03  3d
32b8 : 20 7f 18 20 04 56 ee 7d  4c
32c0 : 32 a9 2c 8d ff 01 20 ac  65
32c8 : 15 8e 82 32 f0 1e ee 82  cc
32d0 : 32 ae 82 32 ec 81 32 90  05
32d8 : 10 ec 7f 32 90 08 f0 06  9e
32e0 : ce 82 32 4c ec 32 8e 81  a3
32e8 : 32 20 2d c7 20 6f cd 20  64
32f0 : 90 32 a5 d0 05 d1 f0 fa  b5
32f8 : 20 9f cd 20 34 c2 a6 f5  3f
3300 : f0 07 c9 94 f0 25 4c 7d  dd
3308 : 33 c9 14 d0 0e ae 82 32  04
3310 : f0 da ce 81 32 ce 82 32  59
3318 : 4c e9 32 a6 f4 d0 e7 c9  c3
3320 : 11 f0 c9 c9 91 f0 c5 c9  a0
3328 : 20 f0 a3 c9 94 d0 11 ae  54
3330 : 81 32 ec 7f 32 b0 b5 ee  53
3338 : 81 32 ce 82 32 4c ce 32  fb
3340 : c9 1d f0 8a c9 13 d0 0b  b4
3348 : 20 83 32 a2 00 8e 82 32  ee
3350 : 4c ec 32 c9 93 d0 14 20  28
3358 : 83 32 ae 81 32 f0 06 20  d3
3360 : 04 56 ca d0 fa 8e 81 32  ea
3368 : 4c 48 33 c9 9d d0 0e ae  d4
3370 : 82 32 f0 58 ce 82 32 ce  ba
3378 : 82 32 4c ce 32 c9 0d f0  88
3380 : 51 2c 80 32 10 f4 ae 7e  ae
3388 : 32 dd 2c 1c f0 ec bc 2c  f9
3390 : 1c c0 9d d0 08 c9 20 90  fe
3398 : 30 c9 41 90 dd c0 1d d0  09
33a0 : 08 c9 30 90 24 c9 3a 90  46
33a8 : d1 c0 13 d0 10 c9 41 90  2e
33b0 : 18 c9 5b 90 c5 c9 bb 90  50
33b8 : 10 c9 bf 90 bd c0 93 d0  80
33c0 : 08 c9 c1 90 04 c9 de 90  5a
33c8 : b1 ca d0 bd 4c ec 32 20  ff
33d0 : 6e c2 ad 7c 32 85 e8 ad  e9
33d8 : 7d 32 85 e9 a9 0d 20 cf  30
33e0 : 33 a2 00 f0 03 20 e5 90  6c
33e8 : c9 0d f0 15 9d 00 02 e8  ca
33f0 : ec 7f 32 90 f0 2c 80 32  11
33f8 : 30 07 e0 a1 90 e7 4c ed  6d
3400 : a5 a9 00 9d 00 02 a2 ff  c8
3408 : a0 01 20 98 55 ad 00 02  0b
3410 : d0 06 20 a2 52 4c 92 52  b6
3418 : 86 45 84 46 20 a9 15 85  d9
3420 : 4b 84 4c a2 01 b5 3d 95  f3
3428 : 4d b5 45 95 3d ca 10 f5  aa
3430 : 20 80 03 24 0f 10 2d a9  4f
3438 : 00 85 09 85 0a a5 3d a4  fa
3440 : 3e 20 b8 15 20 1f 79 20  80
3448 : b2 15 20 86 03 f0 07 c9  c5
3450 : 2c f0 03 4c 64 57 a2 01  cc
3458 : b5 3d 95 45 b5 4d 95 3d  50
3460 : ca 10 f5 60 a2 00 86 13  26
3468 : 4c 3a 57 f0 11 20 f4 87  ba
3470 : ac 2e 0a e0 01 f0 0d e0  60
3478 : 02 f0 0f 4c 28 7d ac 2e  bd
3480 : 0a 98 f0 06 a2 10 88 10  c0
3488 : 06 60 a2 00 88 10 fa ad  b7
3490 : f0 20 d0 f5 24 d7 10 f1  a8
3498 : 8a 49 10 8d 2e 0a 20 02  34
34a0 : 35 ad 2e 0a a2 0c 20 a3  cb
34a8 : 24 18 69 08 8d 2f 0a a2  f3
34b0 : 14 20 a3 24 ad 22 35 f0  e4
34b8 : 08 a5 eb 48 a5 ec 4c 86  97
34c0 : cc ee 22 35 20 24 ca 4c  1a
34c8 : de 6a 20 b3 25 a5 7f 30  38
34d0 : 12 ce 01 35 d0 0d ee 01  64
34d8 : 35 a5 d3 29 08 aa f0 03  99
34e0 : 20 e6 34 4c 65 fa a5 d3  76
34e8 : a2 0a 8e 01 35 a0 10 c9  7f
34f0 : 09 f0 91 a0 00 c9 0a f0  42
34f8 : 91 c9 08 d0 03 4c 7e 34  7f
3500 : 60 0a 48 8a 20 3d 35 a0  cb
3508 : 15 b9 e4 00 20 a1 24 88  e4
3510 : 10 f7 a0 0d b9 54 03 20  70
3518 : a1 24 88 10 f7 68 20 3d  ad
3520 : 35 a2 00 d0 01 60 a0 15  80
3528 : 20 a9 24 99 e4 00 88 10  ea
3530 : f7 a0 0d 20 a9 24 99 54  89
3538 : 03 88 10 f7 60 18 69 07  fd
3540 : 85 17 a9 d0 85 16 4c 3d  8a
3548 : 1e f0 05 20 80 03 d0 fb  7f
3550 : 60 4c 28 7d a2 ff a9 00  61
3558 : f0 09 a2 00 a9 ff d0 03  59
3560 : a9 00 aa 8d 1f 17 8e 4c  e3
3568 : 16 20 f4 87 e0 10 b0 e1  d1
3570 : 86 fe 20 09 88 2c 1f 17  33
3578 : 30 04 e0 10 b0 d3 8e ef  a8
3580 : 03 20 0f 88 84 fa 85 fb  96
3588 : 20 0f 88 c8 d0 03 18 69  c3
3590 : 01 84 fc 85 fd 2c 1f 17  af
```

Listing 1. (Fortsetzung)

```
3598 : 10 0a a6 fe 20 8a 14 86  ea
35a0 : fe 4c 4d 16 20 0f 88 84  80
35a8 : 65 20 45 a8 a6 fe 20 8a  7c
35b0 : 14 8e 0a 16 8e c1 15 ae  f9
35b8 : ef 03 20 8a 14 8e 11 16  a8
35c0 : 8e d8 15 38 a5 fa e5 65  9b
35c8 : a5 fb e5 66 a8 30 03 4c  62
35d0 : 07 16 a5 65 18 65 fc a8  ea
35d8 : a5 66 65 fd aa 98 38 e5  e5
35e0 : fa 85 65 8a e5 fb 85 66  69
35e8 : 4c be 15 a0 02 a5 0f d0  18
35f0 : 07 88 a5 10 d0 02 a0 04  4e
35f8 : 60 20 4c 36 20 a9 15 8d  01
3600 : e4 03 8c e5 03 68 85 63  96
3608 : 68 85 62 68 85 61 20 eb  94
3610 : 35 c4 63 f0 03 4c e7 77  bf
3618 : c8 8c e6 03 20 49 36 a5  b1
3620 : 61 cd e4 03 d0 08 a5 62  aa
3628 : cd e5 03 d0 01 60 ad e6  5a
3630 : 03 18 65 61 85 61 90 e4  34
3638 : e6 62 4c 1c 36 20 a9 15  1b
3640 : 85 61 84 62 20 eb 35 84  22
3648 : 63 4c 71 16 20 a9 15 8d  af
3650 : 72 36 68 8d 70 36 68 8d  1f
3658 : 71 36 ad 72 36 48 98 48  37
3660 : 20 eb 35 98 48 ad 71 36  fa
3668 : 48 ad 70 36 48 4c 5c 79  b5
3670 : 00 00 00 a9 ff d0 02 a9  87
3678 : 00 8d 1f 17 20 4c 36 20  67
3680 : a9 15 20 4f 36 20 a9 15  db
3688 : 85 61 84 62 20 eb 35 84  6a
3690 : 63 8c e6 03 c8 8c e9 03  f2
3698 : 68 c5 63 f0 03 4c e7 77  fb
36a0 : 68 8d eb 03 68 8d ea 03  cf
36a8 : 68 c5 63 d0 f0 68 85 64  17
36b0 : 68 85 63 20 17 37 a5 63  40
36b8 : cd ea 03 d0 07 a5 64 cd  a0
36c0 : eb 03 f0 1a ad e9 03 aa  38
36c8 : 18 65 61 85 61 90 02 e6  0c
36d0 : 62 8a 18 65 63 85 63 90  3b
36d8 : 02 e6 64 4c b3 36 60 a9  b2
36e0 : ff d0 02 a9 00 8d 1f 17  14
36e8 : 20 4c 36 20 a9 15 85 61  dc
36f0 : 84 62 8d e7 03 8c e8 03  44
36f8 : 68 8d e6 03 68 8d e5 03  d1
3700 : 85 64 68 8d e4 03 85 63  c6
3708 : 20 eb 35 cc e6 03 f0 03  55
3710 : 4c e7 77 c8 8c e9 03 4c  03
3718 : a7 16 20 f4 87 e0 01 f0  d6
3720 : 05 90 2e 4c 28 7d 20 0f  90
3728 : 88 c9 20 90 f6 c5 30 d0  af
3730 : 04 c4 2f f0 a9 c5 3a 90  53
3738 : 06 d0 13 c4 39 b0 0f 85  64
3740 : 30 84 2f 85 32 84 31 85  46
3748 : 34 84 33 4c fa 51 4c 3a  f5
3750 : 4d 20 0f 88 c9 40 90 cb  fb
3758 : d0 04 c0 00 f0 c5 ad 10  6e
3760 : 12 38 e5 2d a8 ad 11 12  0e
3768 : e5 2e 18 aa 98 65 16 90  ee
3770 : 01 e8 a8 8a 18 65 17 b0  cb
3778 : d5 cd 13 12 90 07 d0 ce  5d
3780 : cc 12 12 b0 c9 20 92 2a  2c
3788 : 8d 89 2a 8d ef 03 85 fe  41
3790 : 8d 4c 16 8d 1f 17 a5 16  e8
3798 : a6 17 85 65 86 66 8d ea  7f
37a0 : 03 8e eb 03 a5 2d a6 2e  00
37a8 : 85 fa 86 fb 8d ec 03 8e  35
37b0 : ed 03 ad 10 12 85 fc ad  29
37b8 : 11 12 85 fd 20 a9 35 ad  73
37c0 : ea 03 ae eb 03 85 2d 86  73
37c8 : 2e 85 fa 86 fb a4 fa d0  bb
37d0 : 02 c6 fb c6 fa a0 00 98  f3
37d8 : 91 fa a5 2d 38 ed ec 03  a2
37e0 : 85 fc 8a ed ed 03 85 fd  4d
37e8 : a8 a5 fc 18 aa 6d 10 12  1f
37f0 : 8d 10 12 98 6d 11 12 8d  e0
37f8 : 11 12 8a 18 65 43 85 43  c5
3800 : 98 65 44 85 44 24 7f 10  90
3808 : 3d 8a 18 65 3d 85 3d 98  63
3810 : 65 3e 85 3e 8a 18 6d 02  e1
3818 : 12 8d 02 12 98 6d 03 12  d9
3820 : 8d 03 12 8a 18 6d 0e 12  4e
3828 : 8d 0e 12 98 6d 0f 12 8d  07
3830 : 0f 12 a5 7d 85 3f a5 7e  47
3838 : 85 40 a5 3f c9 ff d0 09  20
3840 : a5 40 c9 09 d0 03 4c fe  ed
3848 : 2a a0 00 b1 3f c9 81 d0  e2
3850 : 18 a0 0f 20 5a 38 a9 12  b2
3858 : d0 16 b1 3f 18 65 fc 91  4b
3860 : 3f c8 b1 3f 65 fd 91 3f  63
3868 : 60 a0 03 20 5a 38 a9 05  f5
3870 : 18 65 3f 85 3f 90 c3 e6  11
3878 : 40 d0 bf 20 f4 87 e0 01  25
3880 : f0 22 90 06 4c 28 7d 4c  fb
3888 : 3a 4d 20 0f 88 cd 11 12  b2
3890 : 90 f5 d0 05 cc 10 12 90  a6
3898 : ee c9 ff b0 e7 8d 13 12  dc
38a0 : 8c 12 12 60 20 0f 88 c5  ee
38a8 : 30 90 dc d0 04 c4 2f 90  b6
38b0 : d6 c9 ff f0 cf 85 3a 84  a4
38b8 : 39 a9 00 4c f8 51 20 f0  cc
38c0 : 84 ad 10 12 8d 4d 3a ad  e9
38c8 : 11 12 8d 4e 3a 20 12 88  0d
38d0 : 8d 48 3a 8c 47 3a 20 0f  86
38d8 : 88 cd 48 3a 90 07 d0 08  35
38e0 : cc 47 3a b0 03 4c 28 7d  23
38e8 : 8d 4a 3a 8c 49 3a 20 0f  bf
38f0 : 88 8d 4c 3a 8c 4b 3a cd  41
38f8 : 4a 3a 90 07 d0 11 cc 49  c0
3900 : 3a b0 0c cd 48 3a 90 07  f6
3908 : d0 db cc 47 3a b0 d6 20  a7
3910 : 06 9e b0 02 a0 0a 8c 70  3f
3918 : 11 8d 71 11 20 06 9e b0  7c
3920 : 02 a0 0a 8c 72 11 8d 73  53
3928 : 11 0d 72 11 f0 b7 ad 49  95
3930 : 3a 85 16 ad 4a 3a 85 17  23
3938 : 20 64 50 b0 51 f0 09 38  e6
3940 : a0 01 20 94 50 4c 3b 39  c2
3948 : ad 49 3a d0 08 ad 4a 3a  ce
3950 : f0 18 ce 4a 3a ce 49 3a  fd
3958 : ad 4a 3a cd 48 3a 90 0a  1f
3960 : d0 cc ad 49 3a cd 47 3a  ce
3968 : b0 c4 ad 4d 3a ae 4e 3a  56
3970 : 8d 10 12 8e 11 12 ca 85  34
3978 : fb 86 fc a0 fd a9 00 91  5a
3980 : fb c8 d0 fb 85 5c 85 5d  9f
3988 : 20 fe 2a 4c 68 5b ad 4a  e8
3990 : 3a cd 48 3a 90 d4 d0 08  0d
3998 : ad 49 3a cd 47 3a 90 ca  50
39a0 : ad 49 3a d0 03 ce 4a 3a  df
39a8 : ce 49 3a a5 62 85 25 a5  90
39b0 : 61 85 24 a0 00 20 ec 42  2a
39b8 : 38 e5 61 85 fc 20 23 3a  be
39c0 : 20 af 5d a4 fc 84 0d 20  0c
39c8 : c6 5d 20 fe 2a ad 4b 3a  d6
39d0 : 85 16 ad 4c 3a 85 17 a9  d5
39d8 : 00 8d 1f 17 20 64 50 b0  11
39e0 : 1b f0 0c a0 01 8c 1f 17  aa
39e8 : 38 20 94 50 4c df 39 ee  e6
39f0 : 4b 3a d0 03 ee 4c 3a ad  82
39f8 : 1f 17 d0 d1 a5 61 85 24  d5
3a00 : a5 62 85 25 20 af 5d a4  1b
3a08 : fc 84 6d 20 df 5d 20 36  7c
3a10 : 3a 20 fe 2a ad 49 3a 8d  88
3a18 : 4b 3a ad 4a 3a 8d 4c 3a  eb
3a20 : 4c 2e 39 a4 fc 88 20 ec  d5
3a28 : 42 99 2d 1b 88 d0 f7 20  15
3a30 : ec 42 99 2d 1b 60 a4 fc  8b
3a38 : 88 b9 2d 1b 91 61 88 d0  33
3a40 : f8 b9 2d 1b 91 61 60 00  69
3a48 : 00 00 00 00 00 00 00 20  89
3a50 : 81 14 8d 00 40 8d 01 40  34
3a58 : 8d 02 40 85 2f 85 31 85  96
3a60 : 33 a9 20 85 30 85 32 85  24
3a68 : 34 a9 40 85 2e 8d 11 12  e9
3a70 : a9 01 85 2d a9 03 8d 10  aa
3a78 : 12 20 24 ca a0 00 b9 0e  0a
3a80 : 3b c9 40 d0 0e 24 d7 10  4b
3a88 : 0d a2 13 a9 20 20 69 92  ae
3a90 : ca d0 f8 20 69 92 c8 c0  d4
3a98 : 56 d0 e3 20 a1 3a 4c f8  62
3aa0 : 51 20 64 3b a9 3f 20 af  f6
3aa8 : 02 88 a2 7f a9 d8 20 af  c8
3ab0 : 02 8c 00 ff 84 76 84 d8  b8
3ab8 : 8c 6a 11 8c f0 20 ad 06  22
3ac0 : d5 29 f0 09 06 8d 06 d5  18
3ac8 : a9 05 8d 00 0a a9 3c 8d  51
3ad0 : 01 0a a9 00 8d 12 12 a9  46
3ad8 : bf 8d 13 12 a9 2f 8d 2e  0c
3ae0 : 03 a9 3c 8d 2f 03 78 a2  ab
3ae8 : 19 bd f4 3a 9d fc 02 ca  c4
3af0 : 10 f7 58 60 ed 3b 00 00  d7
3af8 : 67 19 48 3c 0d 43 74 2e  9f
3b00 : 7b 3c bc 19 76 3b 9d 3b  1a
3b08 : db 3b 00 00 ca 34 0d 13  2a
3b10 : 13 0b 0e 05 93 40 20 20  c9
3b18 : d3 55 50 52 41 20 c2 41  97
3b20 : 53 49 43 20 2b 0d 40 20  49
3b28 : 20 3c 43 3e 20 42 59 20  b9
3b30 : d4 ce d4 20 2d 20 d3 4f  66
3b38 : 46 54 57 4f 52 4b 0d 40  9c
3b40 : 20 20 38 39 35 39 37 20  e0
3b48 : c2 41 53 49 43 20 c2 59  9c
3b50 : 54 45 53 20 46 52 45 45  b6
3b58 : 20 32 30 2e 30 38 2e 31  43
3b60 : 39 38 38 0d a9 ff 85 fe  14
3b68 : a9 f8 85 fd a9 fd 8d b9  e2
3b70 : 02 a0 01 a2 7f 60 85 fc  62
3b78 : a9 3c a0 cb 20 e2 43 90  28
3b80 : 0f a2 00 29 7f 18 69 40  e4
3b88 : 4c b2 43 a5 fc 4c 26 43  04
3b90 : a9 3e a0 4f 20 e2 43 90  b2
3b98 : f2 a2 ff d0 e6 20 a3 3b  6a
3ba0 : 4c 2e 51 29 7f 38 e9 40  5f
3ba8 : 09 80 e8 f0 27 aa a9 3c  30
3bb0 : a0 cb 85 25 84 24 a0 00  28
3bb8 : ca 10 0f b1 24 48 e6 24  ed
3bc0 : d0 02 e6 25 68 10 f4 30  2b
3bc8 : ef c8 b1 24 30 05 20 0c  d0
3bd0 : 56 d0 f6 60 aa a0 4f a9  98
3bd8 : 3e d0 d7 29 7f 38 e9 40  7b
3be0 : 0a a8 b9 f9 3e 48 b9 f8  eb
3be8 : 3e 48 4c 80 03 29 3f 0a  f8
3bf0 : a8 b9 89 3f 8d ff 3b b9  f8
3bf8 : 88 3f 8d fe 3b 20 00 00  18
3c00 : 20 56 79 18 60 ae f0 20  2c
3c08 : f0 03 20 40 1f 20 a1 3a  78
3c10 : 20 7d ff 0b 0e 05 00 4c  f2
3c18 : 03 40 20 27 32 a5 61 a4  47
3c20 : 62 18 6d fd 31 90 01 c8  d7
3c28 : 85 3d 84 3e 4c f6 4a c9  6e
3c30 : 42 f0 03 4c 06 b0 20 b4  05
3c38 : b8 a2 00 a0 02 8c 49 3c  78
3c40 : d0 14 8a 30 d5 4c 00 b0  e4
3c48 : a2 00 f0 04 a0 00 f0 ed  51
3c50 : ca 86 3c 20 93 4f 86 3d  b9
3c58 : 84 3e 20 80 03 aa f0 e8  2f
3c60 : b0 03 4c e2 4d 20 0a 43  86
3c68 : 20 86 03 c9 de f0 03 4c  df
3c70 : f3 4a 20 9f 3c 20 86 03  69
3c78 : 4c f6 4a 08 20 80 03 c9  79
3c80 : de f0 0c 20 c2 3c 28 4c  25
3c88 : a2 4a a9 00 8d 93 3c 20  60
3c90 : 9f 3c a9 00 d0 03 20 c2  e3
3c98 : 3c 20 c2 3c 4c a2 4a 20  60
```

```
3ca0 : 80 03 c9 9c d0 03 4c 6c  d7
3ca8 : 79 20 f4 87 e0 01 f0 0c  51
3cb0 : 90 07 e0 05 f0 03 4c 28  45
3cb8 : 7d 4c 6c 6b ee 93 3c 4c  f9
3cc0 : 05 1f a5 3d d0 02 c6 3e  1b
3cc8 : c6 3d 60 56 4c 4f 43 41  df
3cd0 : 54 c5 4f 4c c4 5a 4b d2  56
3cd8 : 4f 46 c6 56 42 4f d8 53  6f
3ce0 : 43 52 45 45 ce 56 43 4f  92
3ce8 : 4c 4f d2 47 52 4f ce 47  e3
3cf0 : 52 43 4c d2 47 52 4f 46  22
3cf8 : c6 57 56 52 45 c7 56 44  be
3d00 : 52 41 d7 56 43 49 52 43  02
3d08 : 4c c5 56 44 43 53 41 56  d5
3d10 : c5 56 44 43 4c 4f 41 c4  48
3d18 : 44 50 4f 4b c5 4c 57 49  70
3d20 : 4e c4 54 4b 45 d9 58 4b  6a
3d28 : 45 d9 46 49 4e c4 4d 45  df
3d30 : 52 47 45 b2 4d 45 52 47  a4
3d38 : c5 4c 4c 49 53 d4 4c 53  13
3d40 : 45 d4 4c 4f 43 4b d3 55  75
3d48 : 4e 4c 4f 43 4b d3 43 55  04
3d50 : 4f ce 43 4f 46 c6 43 47  f7
3d58 : 4f 54 cf 43 47 4f 53 55  14
3d60 : c2 4f 46 49 4e 50 55 d4  eb
3d68 : 43 4c 49 53 d4 57 4b 45  4e
3d70 : d9 45 52 41 53 c5 4b 49  cc
3d78 : 4c cc 43 4f 44 c5 44 45  f3
3d80 : 43 4f 44 c5 43 45 4e 54  75
3d88 : 52 c5 52 45 53 45 d4 50  4d
3d90 : 41 55 53 c5 44 49 4e 4c  6a
3d98 : 4f 43 cb 44 49 4e 46 52  c9
3da0 : 45 c5 53 54 52 56 44 c3  98
3da8 : 45 58 43 48 41 4e 47 c5  22
3db0 : 45 58 46 49 ae 43 4c 45  9d
3db8 : 41 d2 43 4c 46 49 ae 54  cf
3dc0 : 52 41 4e 53 46 45 d2 54  33
3dc8 : 52 41 4e 46 49 ae 44 55  dd
3dd0 : 4d d0 50 41 47 c5 50 52  4a
3dd8 : 4f c3 4d 4f 56 c5 46 49  85
3de0 : 4c cc 53 45 54 42 4f d4  4e
3de8 : 53 45 54 54 4f d0 56 43  d9
3df0 : 48 41 d2 4d 45 4d 53 57  f2
3df8 : 41 d0 44 49 53 cb 44 49  13
3e00 : 53 45 d4 52 45 4c 4f 43  f0
3e08 : 41 54 c5 45 58 45 43 55  f5
3e10 : 54 c5 4c 50 52 49 4e d4  b6
3e18 : 49 4e 50 55 54 46 4f 52  a0
3e20 : cd 56 50 41 49 4e d4 53  56
3e28 : 54 52 4d 45 cd 48 49 44  6e
3e30 : c5 55 4e 49 54 c5 57 52  d2
3e38 : 55 53 d2 52 45 50 45 41  a4
3e40 : d4 53 45 54 54 49 4d c5  ea
3e48 : 58 53 54 56 44 c3 00 49  1f
3e50 : 4e 53 43 d2 56 52 44 4f  1b
3e58 : d4 52 56 52 45 c7 4c 4f  98
3e60 : d7 48 49 47 c8 44 50 45  11
3e68 : 45 cb 52 42 41 4e cb 55  d0
3e70 : 53 45 c4 52 41 c4 44 45  b7
3e78 : c7 43 4f d4 4c c7 52 44  24
3e80 : 4c 49 4e c5 44 55 d0 52  94
3e88 : 42 4f d4 52 54 4f d0 50  95
3e90 : 52 47 4c 49 4e c5 44 49  79
3e98 : d6 4d 4f c4 46 52 41 c3  05
3ea0 : 49 4e 53 45 52 d4 41 d4  08
3ea8 : 49 4e 4b 45 d9 4d 45 4d  4c
3eb0 : 53 54 d2 42 41 53 49 c3  86
3eb8 : 42 41 53 4c ce 44 49 4e  ca
3ec0 : 50 55 d4 54 49 4d c5 45  1b
3ec8 : 56 41 cc 4d 49 ce 4d 41  5e
3ed0 : d8 4f 44 c4 52 4f 55 4e  8b
3ed8 : c4 56 44 43 53 54 d2 41  e6
3ee0 : 53 43 42 c9 42 49 41 53  b9
3ee8 : c3 43 55 d4 4c 43 55 d4  1b
3ef0 : 44 45 c3 44 45 43 cc 00  f2
3ef8 : b0 24 e5 2a 7e 1e 64 25  2e
3f00 : 81 23 2c 1e 63 24 16 1f  d0
3f08 : 19 1f 3f 1f 57 1f 6e 1f  cb
3f10 : c2 22 e8 24 0b 25 3b 29  bb
3f18 : 62 29 93 29 ac 29 e9 29  27
3f20 : 87 2a a7 2a b6 2d ed 2d  d2
3f28 : 2f 30 35 30 08 2b 0e 2b  2b
3f30 : a9 2f 23 30 a0 32 69 2e  dd
3f38 : 3e 1f 3d 2e 20 2e 97 30  4d
3f40 : e6 30 0a 19 3c ff fc 26  e8
3f48 : 54 27 61 27 9d 28 e2 36  80
3f50 : 76 36 3c 36 f8 35 de 36  d8
3f58 : 72 36 02 13 6a 34 48 35  9c
3f60 : 5f 35 59 35 19 37 7a 38  fd
3f68 : 71 2b 53 35 5b 31 98 31  ee
3f70 : bd 38 a7 31 b2 2d 34 32  23
3f78 : 1f 17 f1 2c b4 2a 87 2a  34
3f80 : 81 2d 8b 2d 3c 26 99 28  cc
3f88 : 6c 1e bc 24 5a 1e 4d 1e  bf
3f90 : 54 1e 1f 29 1b 2e 01 2e  64
3f98 : 8a 30 7b 30 5a 30 6a 30  50
3fa0 : 4c 30 d8 13 42 31 23 31  39
3fa8 : 53 30 34 2b 30 2b 65 2b  ce
3fb0 : 17 15 dc 18 e9 26 9c 14  f6
3fb8 : 0a 2d 6b 2d 58 2f ac 26  d7
3fc0 : 13 27 6d 27 69 27 b8 27  a8
3fc8 : ca 27 1c 28 48 28 5e 28  c2
3fd0 : c6 28 fa 28 de 28 e9 28  95
3fd8 : a2 ff 78 9a d8 20 42 e2  49
3fe0 : 20 09 e1 20 93 e0 20 56  6f
3fe8 : e0 20 00 c0 20 7a 41 20  0c
3ff0 : 45 40 20 4f 3a 4c 3c 40  bf
3ff8 : 00 08 f7 00 f7 00 f7 21  9c
```

Listing 1. (Schluß)

Power 128 ist ein mächtiges Werkzeug für den C128 mit der Floppy 1571. Nahezu alle erdenklichen Manipulationen im Directory einer Diskette lassen sich mit diesem Programm durchführen.

POWER 128 -DAS SUPER-TOOL

Power 128 ist ein reinrassiges Maschinenspracheprogramm, das ein Höchstmaß an Komfort bei der Arbeit mit Disketten bietet. Es hat eine exzellente Benutzerführung durch Menüs, die mit den Funktionstasten aufgerufen werden, arbeitet im 80-Zeichen-Modus des C128 und zeigt ständig das aktuelle Laufwerk und Ausgabegerät in der Titelleiste an.

Power 128 (Listing 1) ist sowohl allein lauffähig als auch von einem Basic-Programm aufrufbar. Das Maschinenprogramm belegt den Speicherbereich von $1300 bis $8908 in Bank 0 und von $EA00 bis $FDFE in Bank 1. Es wird entweder gebootet oder nach dem Laden mit »SYS 4864« gestartet. Durch eine eingebaute Schnittstelle können Sie Power 128 für Ihre persönlichen Bedürfnisse erweitern: Die Tastenkombination <SHIFT RUN/STOP> wird in allen Programmteilen abgefragt, aber nirgends für ein Unterpro-

gramm verwendet. Hier können Sie Ihre eigenen Unterprogramme einbinden (Tabelle 1).

Nach dem Start des Programms mit

BOOT"POWER 128"

erscheint ein Übersichtsmenü, das die Auswahl aus neun Hauptfunktionen mittels der Funktionstasten erlaubt. Ein Druck auf <ESC> beendet das Programm und führt entweder zurück in den Direktmodus oder in das Basic-Programm, von dem Power 128 aufgerufen wurde.

Schnittstellen

$1300	Startadresse
$1711	Einsprungadresse für Basic-Programme
$139F	Adresse des ersten Laufwerks
$13A0	Adresse des zweiten Laufwerks
$19DF - $19EE	Trennzeile
$3783 - $3785	Platzhalter für Erweiterung. Dieser besteht aus drei »NOPs« und wird bei <SHIFT-RUN/STOP> angesprungen (kann beispielsweise mit »jsr xxxx« ausgefüllt werden).

Tabelle 1. Hier finden Sie die wichtigsten Schnittstellen

Detaillierte Beschreibung der Funktionen

Nahezu alle Optionen des Programms werden über die Funktionstasten aufgerufen. Diese sind:

<F1> Extended Directory

In diesem Menü stehen folgende Funktionen zur Verfügung: Mit <F7> kann zwischen den beiden voreingestellten Diskettenstationen umgeschaltet werden, mit <F8> die Ausgabe auf den Monitor oder auf einen angeschlossenen Drucker gelenkt, mit <HELP> das »Standard-Directory« ausgegeben oder mit <ESC> die Routine verlassen und ins Hauptmenü zurückgesprungen werden. Mit <F1> startet man nun die Ausgabe des ausführlichen Directory. Folgende Disketten-Daten werden ausgegeben:
- Header
- ID
- Disk-Format
- Anzahl Boot-Blöcke
- Disk-Status (Read Only, Read/Write)
- Anzahl Directory-Einträge
- Anzahl freie Blöcke (laut BAM)
- Anzahl belegter Blöcke (Summe der File-Längen)
- Gesamt-Kapazität der Diskette

Zusätzlich wird bei der Druckerausgabe noch die Eingabe des aktuellen Datums oder einer Kommentar-Zeile angefordert. Für jeden Directory-Eintrag werden folgende Daten ermittelt und ausgegeben:
- File-Name
- File-Typ
- File-Status
- Anzahl belegter Blöcke
- Track und Sektor des ersten Blocks jedes Files
- Start-Adresse aus File-Header
- Endadresse (maximal)

Bei REL-Files wird zusätzlich die Länge des Datensatzes angegeben. Die Ausgabe kann durch einen kurzen Druck auf die Space-Taste angehalten und durch eine weitere Betätigung von <SPACE> fortgesetzt werden.

Sind alle Einträge analysiert und ausgegeben, kann die nächste Diskette eingelegt und mit <F1> der Ablauf wiederholt werden. <ESC> beendet das Unterprogramm und führt zurück ins Hauptmenü.

<F2> Save and restore Directory Tracks

Besonders bei professioneller Software ist es empfehlenswert, vor Manipulationen des Directory, wie Sortieren, File-Status-Manipulationen etc. die komplette Directory-Spur in einem externen File zu sichern. Diese Funktion wird hier angeboten:

<F1> Einlesen der Directory-Spur in den Speicher
<F3> Einlesen eines Directory-Save-Files in den Speicher
<F4> Speicherinhalt in Directory-Spur schreiben
<F6> Speicherinhalt in Directory-Save-File schreiben

Demzufolge wird eine Directory-Spur durch <F1> und <F6> gesichert sowie durch <F3> und <F4> restauriert

Des weiteren stehen noch folgende Funktionen zur Verfügung:

<F7> Umschalten zwischen verschiedenen Diskettenlaufwerken
<HELP> Standard-Directory
<ESC> zurück ins Hauptmenü

<F3> Modify BAM & Directory

Folgende Unterfunktionen stehen zur Auswahl:

<F1> BAM modifizieren
<F2> Diskette formatieren
<F3> Directory-Einträge modifizieren
<F4> Diskette validieren (Boot-Sektor wird berücksichtigt).
<F7> Laufwerk wechseln
<HELP> Standard-Directory

Die einzelnen Funktionen von <F1> (BAMmodifizieren) sind folgendermaßen zu bedienen:

<F1> Header der eingelegten Diskette ändern
<F2> Summe freier Blöcke auf 0 setzen
<F3> ID ändern (maximal fünf Zeichen!)
<F4> Summe freier Blöcke auf Maximalwert setzen
<+> nächste Bildschirmseite
<U> Update (Änderungen auf Diskette zurückschreiben)
<ESC> Zurück zum aufrufenden Untermenü
<F2> (Diskette formatieren) sendet den Formatbefehl des DOS an die Floppy. Vorher werden Diskname und ID abgefragt.
<F3> (Directory-Einträge modifizieren) stellt Directory-Einträge dreispaltig und farbcodiert dar. Die Voreinstellungen der Farben können anhand der POKE-Liste in Tabelle 2 modifiziert werden. Die Farbgebung wird durch den Filetyp bestimmt; voreingestellt sind folgende Werte:

Farben

$1363	Textfarbe
$1366	Kopfzeile
$1369	Standard
$136C	Fehlermeldungen
$136F	Anweisungen
$1372	Mitteilungen
$2B63, $2B6F	DEL-Files
$2B64, $2B7C	SEQ-Files
$2B65, $2B88	PRG-Files
$2B66, $2B94	USR-Files
$2B67, $2BA0	REL-Files
$53AC	Menüleisten

Tabelle 2. In diesen Speicherstellen sind die Farben abgelegt

DEL	Rot
SEQ	Braun
PRG	Gelb
USR	Grün
REL	Blau

Folgende Funktionen stehen zur Auswahl:

<F1> Directory alphabetisch sortieren
<F2> File-Länge ausgewählter Files = 0
<F3> Extension »,U{Laufwerk}« an einen Filenamen hängen. Dies dient zum bequemeren Laden von Programmen.
<F4> File-Länge ausgewählter Files auf den Wert 65535 setzen.
<F5> beliebige Extension an einen Filename hängen
<+> nächste Bildschirm-Seite anzeigen, falls nicht alle Einträge auf einen Screen passen.
<U> Update, Änderungen werden gespeichert
<ESC> zurück zum aufrufenden Untermenü.

Ein Beispiel zur Funktion <F5>:

Um eine beliebige Extension an einen oder mehrere Filenamen anhängen zu können, wird man nach <F5> zunächst aufgefordert, in einem einzeiligen Eingabefenster die gewünschte Extension einzugeben (Abschluß mit RETURN). Ist die Extension definiert, wird man aufgefordert, die Kennzahlen des oder der Files einzugeben, die diese Extension erhalten sollen. Diese Eingabe kann also für nur ein File oder auch für einen ganzen File-Block erfolgen. Soll eine Extension (beispielsweise »,8«) an mehrere Files angehängt werden, so sind die erste und letzte Nummer des Fileblocks durch Bindestrich getrennt anzugeben.

Ist die Eingabe korrekt , wird die Extension angehängt, wenn nicht, wird eine neuerliche Eingabe verlangt. Durch Eingabe von »0« kann man die File-Auswahl jederzeit abbrechen.

<F4> Diskette validieren (Boot-Sektor wird berücksichtigt!)
<F7> Umschalten zwischen verschiedenen Diskettenlaufwerken
<HELP> Standard-Directory

<F4> Protect and Release Disks & Files

Setzt oder löscht den Schreibschutz für ganze Disketten durch Ändern des Disk-Format-Kennbytes und für einzelne Files durch Änderung des Typ-Bytes. Damit ist es möglich, das leidige DOS-Problem der 1571 ohne Änderung des DOS-ROMs und ohne Verwendung eines Diskmonitors zu umgehen.

<F1> Diskette schützen
<F2> Diskettenschreibschutz löschen
<F3> Files schützen
<F4> Files freigeben
<F7> Disketten-Station wechseln
<HELP> Directory
<ESC> zurück ins Hauptmenü

Bei den Funktionen <F1> und <F2> genügt es, die Anweisungen auf dem Bildschirm zu befolgen. Bei F3 und F4 wird zunächst die Directory-Spur eingelesen und der Inhalt in gewohnter Form auf dem Bildschirm angezeigt. Nun kann man die zu schützenden oder die freizugebenden Files anwählen und nach Abschluß des Auswahl-Vorgangs mit »U« die Veränderungen auf Diskette zurückschreiben.

<F5> Sort & Modify Directory-Entries

Zunächst hat man die Wahl, den Sortier- und Modifizier-Vorgang zu starten, mit <F7> die Disketten-Station zu wechseln, mit <HELP> das Directory auf den Bildschirm zu holen oder mit <ESC> zurück ins Hauptmenü zu springen.

Entscheidet man sich dafür, mit <SPACE> den nächsten Menüpunkt anzuwählen, so wird man aufgefordert, entweder die Diskette einzulegen und <SPACE> zu drücken oder die Aktion mit <ESC> abzubrechen. Nach dem Einlesen der Diskette hat man folgende Möglichkeiten:

<F1> Files oder File-Blöcke verschieben
<F2> Directory-Einträge entfernen (mit Sicherheitsabfrage)
<F3> Trennzeile einfügen
<F5> Directory alphabetisch sortieren
<F6> Directory nach der File-Länge sortieren
<U> Update (Änderungen auf Diskette zurückschreiben)
<ESC> zurück ins aufrufende Submenü

Die Syntax bei File-Verschiebungen lautet bei Einzel-File-Verschiebungen:

<ziffer1>-<ziffer2>

Das File auf Position »ziffer1« wird von seiner momentanen Position nach »ziffer2« verschoben.

Bei Block-Verschiebungen ist die Syntax folgendermaßen:

<ziffer1>-<ziffer2>-<ziffer3>

Dabei werden die Files aus dem Bereich »ziffer1« bis »ziffer2« nach »ziffer3« verschoben.

Löschungen werden nur für Einzelfiles akzeptiert, um unbeabsichtigtes Löschen zu vermeiden.

<F6> Directory selektiv ausdrucken

Mit dieser Funktion kann man ein Directory platzsparend ausdrucken. Man hat dabei sogar die Möglichkeit, die benötigten Einträge zu selektieren, bevor diese an den Drucker geschickt werden. Die zur Verfügung stehenden Einstiegsfunktionen sind:

<F7> Laufwerk wechseln
<HELP> Directory auf den Bildschirm holen
<ESC> zurück ins Hauptmenü

Betätigt man die SPACE-Taste, so gelangt man – nach Einlegen der gewünschten Diskette und Bestätigung mit nochmaligem <SPACE> – in ein Untermenü, das folgende Funktionen zur Verfügung stellt:

<F1> File(s) verschieben
<F2> Einträge aus der Druckliste entfernen
<F3> Trennzeile einfügen
<F5> Remark-Zeile eingeben (60 Zeichen stehen zur Verfügung)
<F7> Ergebnis auf dem Bildschirm zeigen
<F8> Setzt das Ende der Druck-Liste auf eine beliebige Position
<P> Druckvorgang starten
<+> nächste Bildschirmseite anzeigen

Ist der Drucker nicht angeschlossen, erscheint eine Fehlermeldung, die – nach entsprechender Korrektur – mit SPACE zu quittieren ist.

<F7> Disk-Labels drucken

Mit dieser Funktion kann man sich bequem Disk-Etiketten zusammenstellen und drucken. Zu Anfang holen Sie sich – wie unter <F6> beschrieben – das Inhaltsverzeichnis der gewünschten Diskette. Nach dem Einlesen der Diskette stehen folgende Funktionen zur Verfügung, die auch in der Menü-Leiste angezeigt werden:

<F1> File(s) verschieben
<F2> Einträge entfernen
<F3> Trennzeile einfügen
<F5> Remark-Zeile einfügen (45 Zeichen)
<F7> bringt das Etikett auf den Bildschirm
<F8> Kopfzeile eingeben
<+> weiterblättern
<P> Druckvorgang einleiten
<ESC> zurück ins aufrufende Untermenü

Für jedes Diskettenlabel stehen drei Spalten mit je elf Zeilen zur Verfügung, insgesamt sind also bis zu 33 Einträge möglich. Dazu sollten Sie aber wissen, daß für jede Trennzeile (<F3>) und für jede Kommentarzeile (<F5>) eine ganze Zeile – der Platz von drei Einträgen – benötigt

wird. Die Kopfzeile (<F8>) ist so aufgebaut, daß die letzten sechs Zeichen in doppelter Höhe und Breite gedruckt werden.

Die Druckeranpassung ist derzeit auf den Epson-FX85 abgestimmt, kann aber anhand der POKE-Liste in Tabelle 4 auf jeden anderen Drucker eingestellt werden. Falls Sie die Parameter der Etiketten verändern wollen, verwenden Sie bitte Tabelle 3.

<F8> Boot-Help

Dieser Menüpunkt ist eine große Hilfe bei der Erstellung von Boot-Sektoren. Es handelt sich um die erste veröffentlichte Routine, die es erlaubt, Assembler-Programme von bis zu 16 Blöcken Länge auf die Bootspur zu legen.

Nachdem Sie <F8> gedrückt haben, steht Ihnen ein Untermenü mit folgenden Funktionen zur Verfügung:

<F1> Boot-Sektor(en) generieren
<F2> Boot-Sektor(en) kopieren
<F4> Boot-Sektor löschen
<F8> Diskette formatieren
<F7> Laufwerk umschalten
<HELP> Directory anzeigen
<ESC> zurück ins Hauptmenü

Die einzelnen Menüpunkte sind – mit Ausnahme der Sektor-Generierung (<F1>) – durch die Benutzerführung weitgehend selbsterklärend. Wenn Sie mit <F1> einen neuen Boot-Sektor generieren, öffnet sich ein neues Untermenü mit folgenden Punkten:

<F1> stellt verschiedene Boot-Varianten zur Verfügung
<F3> Assemblerprogramm in Boot-Sektoren eintragen
<F5> Assemblerprogramm booten
<F7> Laufwerk umschalten
<HELP> Directory anzeigen
<ESC> zurück zum aufrufenden Menü

Falls Sie sich für <F1> entschieden haben, werden Sie aufgefordert, die Diskette einzulegen, auf der der Boot-Sektor erzeugt werden soll. Danach geben Sie die Meldung ein, die während des Boot-Vorgangs erscheint und wählen das File, welches gestartet werden soll. Außerdem haben Sie noch die Möglichkeit, eine bis zu 60 Zeichen lange Zeile mit Basic-Befehlen einzugeben, wobei die einzelnen Befehle durch einen Doppelpunkt getrennt sein müssen. Diese Befehle werden nach Abschluß des Boot-Vorgangs – aber noch vor dem eigentlichen Programmstart – ausgeführt.

Wollen Sie statt eines Maschinenprogramms ein Basic-

Etiketten

$1F41	Anzahl Spalten
$2042	Gesamtlänge des Etiketten-Header
$2043	Länge des Etiketten-Header (links)
$2044	Länge des Etiketten-Header (rechts)
$2045	Anzahl Zeilen für Files
$2046	Anzahl Spalten
$2048	Anzahl Zeilen zwischen den Etiketten
$21FE	Anzahl Leerzeichen zwischen den Spalten

Tabelle 3. Mit Hilfe dieser Speicherstellen können Sie das Etikettenformat ändern

```
Name : power - 128/v1.1  1300 6919
----------------------------------
1300 : 4c c2 16 20 24 ca 20 5f  0f
1308 : 13 20 b3 25 20 74 13 20  ef
1310 : 65.13 20 7d ff 20 20 44  c1
1318 : 52 49 56 45 20 23 20 00  e9
1320 : ad 7c 33 20 fb f9 48 8a  a2
1328 : 20 d2 ff 68 20 d2 ff 60  18
1330 : ad 91 13 f0 10 20 7d ff  81
1338 : 50 52 49 4e 54 45 52 20  c6
1340 : 20 00 4c 52 13 20 7d ff  e6
1348 : 4d 4f 4e 49 54 4f 52 20  43
1350 : 20 00 20 77 13 20 36 24  bb
1358 : 20 68 13 20 5f 13 60 a9  d9
1360 : 93 2c a9 97 2c a9 99 2c  35
1368 : a9 9f 2c a9 96 2c a9 9e  d0
1370 : 2c a9 95 2c a9 12 2c a9  8b
1378 : 92 2c a9 0d 2c a9 02 2c  9d
1380 : a9 82 2c a9 0f 2c a9 8f  c3
1388 : 2c a9 07 20 d2 ff 60 00  fd
1390 : 03 00 01 00 00 00 00 00  d4
1398 : 00 00 00 00 00 0c 00 08  09
13a0 : 09 03 05 20 f9 24 20 74  9a
13a8 : 53 20 f1 24 60 20 36 24  34
13b0 : 20 5f 13 ad 7c 33 48 20  bd
13b8 : 2a 28 68 cd 7c 33 f0 6e  cc
13c0 : 20 24 ca 20 b3 25 20 74  77
13c8 : 13 20 65 13 20 7d ff 20  d5
13d0 : 20 44 52 49 56 45 20 23  26
13d8 : 20 00 ad 7c 33 20 fb f9  0b
13e0 : 48 8a 20 d2 ff 68 20 d2  39
13e8 : ff 20 77 13 20 36 24 20  bc
13f0 : b3 25 20 71 13 20 7d ff  94
13f8 : 41 43 54 55 41 4c 20 44  1a
1400 : 52 49 56 45 20 43 48 41  f5
1408 : 4e 47 45 44 20 54 4f 20  f6
1410 : 44 52 49 56 45 20 23 00  7c
1418 : ad 7c 33 20 fb f9 e0 31  4a
1420 : d0 06 48 8a 20 d2 ff 68  c0
1428 : 20 d2 ff 20 68 13 60 20  96
1430 : e8 14 cd 90 13 f0 71 20  66
1438 : 24 ca 20 74 13 20 65 13  46
1440 : a2 00 a0 47 20 c3 25 ad  03
1448 : 90 13 c9 03 d0 10 20 7d  3e
1450 : ff 4d 4f 4e 49 54 4f 52  ad
1458 : 20 20 00 4c 6b 14 20 7d  e5
1460 : ff 50 52 49 4e 54 45 52  86
1468 : 20 20 00 20 77 13 20 36  99
1470 : 24 20 5f 13 20 71 13 20  f9
1478 : 7d ff 4f 55 54 50 55 54  39
1480 : 2d 44 45 56 49 43 45 20  ef
1488 : 43 48 41 4e 47 45 44 20  f9
1490 : 54 4f 20 44 45 56 49 43  cf
1498 : 45 20 23 00 ad 90 13 20  a2
14a0 : fb f9 20 d2 ff 20 68 13  c3
14a8 : 60 a9 01 ac 8f 13 ae 90  20
14b0 : 13 20 ba ff a9 00 20 bd  18
14b8 : ff 20 c0 ff a2 01 20 c9  3e
14c0 : ff b0 04 20 cc ff 60 20  ab
14c8 : 45 24 ad 90 13 20 c7 31  50
14d0 : ad a1 13 8d 90 13 a2 00  f1
14d8 : 8e 91 13 4c de 14 20 cc  26
14e0 : ff a9 01 38 20 c3 ff 60  dc
14e8 : 20 de 14 ad 90 13 48 cd  90
14f0 : a1 13 f0 07 ad a1 13 a2  b1
14f8 : 00 f0 06 ad a2 13 ae 92  4a
1500 : 13 8d 90 13 8e 91 13 20  62
1508 : a9 14 68 60 a9 0a 8d 7b  fa
1510 : 33 a9 1d a2 15 85 89 86  64
1518 : 8a 20 16 30 60 42 2d 50  ab
1520 : 20 31 33 20 30 30 30 00  ef
1528 : 00 00 00 00 a9 30 a2 fe  cd
1530 : 85 87 86 88 ad 81 33 10  ff
1538 : 03 a9 00 2c a9 ff 8d 27  b5
1540 : 15 a9 00 8d 28 15 a0 00  89
1548 : 20 e6 24 91 87 18 e6 87  fa
1550 : 90 02 e6 88 a9 01 91 87  a4
1558 : 20 f1 24 a9 00 a2 fd 85  c7
1560 : fc 86 fd a9 00 a2 ea 85  20
1568 : fe 86 ff a0 01 20 e6 24  b2
1570 : b1 fe 91 87 20 f1 24 18  48
1578 : a9 00 69 e2 8d 29 15 8a  64
1580 : 69 00 8d 2a 15 e6 ff a9  6e
1588 : 00 8d 2b 15 a0 00 20 e6  15
1590 : 24 b1 fe 20 f1 24 8d 27  15
1598 : 15 c8 20 e6 24 b1 fe 91  e5
15a0 : 87 20 f1 24 18 e6 87 90  30
15a8 : 02 e6 88 ee 28 15 ee 2b  5a
15b0 : 15 ad 2b 15 c9 09 d0 2d  8c
15b8 : ad 27 15 f0 06 ce 28 15  fe
15c0 : 4c 85 15 a9 00 a2 fd 85  61
15c8 : fc 86 fd 18 ad 28 15 0a  0e
15d0 : 90 02 e6 fd 85 fc a0 00  9d
15d8 : 98 20 e6 24 91 fc c8 91  06
15e0 : fc 20 f1 24 60 18 ad 29  bd
15e8 : 15 69 20 90 03 ee 2a 15  46
15f0 : 8d 29 15 85 fa ae 2a 15  00
15f8 : 86 fb a0 00 20 e6 24 b1  d1
1600 : fa 20 f1 24 c9 00 d0 03  f1
1608 : 4c ae 15 a9 00 a2 fd 85  3e
1610 : fc 86 fd 18 ad 28 15 0a  56
1618 : 90 02 e6 fd 85 fc a0 00  e5
1620 : ad 29 15 ae 2a 15 20 e6  17
1628 : 24 91 fc e6 fc 8a 91 fc  95
1630 : 20 f1 24 4c ab 15 00 00  3f
1638 : 00 ad 36 16 ae 37 16 85  67
1640 : fc 86 fd 18 ad 38 16 0a  0b
1648 : 48 a9 00 65 fd 85 fd 68  e6
1650 : 18 65 fc 90 02 e6 fd 85  c6
1658 : fc a0 00 20 e6 24 b1 fc  f9
1660 : aa c8 b1 fc 20 f1 24 85  a7
1668 : fd 86 fc 60 8d c1 16 18  63
1670 : 98 6d c1 16 8d c1 16 c8  c3
1678 : 20 e6 24 b1 fc 20 f1 24  2b
1680 : 48 ad 91 13 f0 1d 30 25  69
1688 : 18 68 c9 20 b0 05 a9 2e  81
1690 : 4c ae 16 c9 5f 90 17 c9  5c
1698 : a0 f0 13 c9 e0 f0 0f a9  d3
16a0 : 2e d0 0b 68 85 f4 c9 14  55
16a8 : d0 04 a9 2e 24 68 c9 11  79
16b0 : b0 02 a9 2e 20 d2 ff cc  c4
16b8 : c1 16 d0 bb a9 00 85 f4  ca
16c0 : 60 00 20 b3 77 ae 00 ff  8c
```

Drucken

$138F	Einstellen der Sekundär-Adresse
$1392	Druckertyp ($01: nicht CBM-Drucker, $FF: CBM oder kompatibel)
$139D	Formfeed für Drucker
$13A2	Druckeradresse
Steuerzeichen:	
$2297 - $229B	Elite
$229C - $22A0	Pica
$22A1 - $22A5	Eng ein
$22A6 - $22AA	Eng aus
$22AB - $22AF	Breit ein
$22B0 - $22B4	Fett ein
$22B5 - $22B9	Fett aus
$22BA - $22BE	Doppelt ein
$22BF - $22C3	Doppelt aus
$22C4 - $22C8	Subscript ein
$22C9 - $22CD	Superscript ein
$22CE - $22D2	Sub-/Superscript aus
$22D3 - $22D7	Zeilenvorschub
$22D8 - $22DC	linken Rand setzen
$22DD - $22E1	unteren Rand löschen
$22E2 - $22E6	unteren Rand setzen
$22E7 - $22EB	Unterstreichen ein
$22EC - $22F0	Unterstreichen aus
$22F1 - $22F5	Drucker initialisieren
$22F6 - $22FA	Vorschub um eine halbe Zeile
$22FB - $22FF	Blattlänge setzen

Tabelle 4. Diese Adressen steuern die Druckeranpassung

Programm booten, so ist die einfachste Möglichkeit, dieses von den oben erwähnten Kommandozeilen aus aufzurufen. Nachdem Sie alle Eingaben abgeschlossen haben, werden Sie aufgefordert, eine Diskette einzulegen, die mit der Bootspur beschrieben wird.

Mit <F3> können Sie ein Assemblerprogramm, das Sie vorher gespeichert haben, auf die Bootspur schreiben. Diese Vorgehensweise eignet sich besonders, um ein Programm zu »verstecken«, denn Programme, die auf der Bootspur abgelegt sind, werden nicht im Directory angezeigt, sondern beim Reset ausgeführt.

<HELP> Standard-Directory ausgeben

Natürlich ist auch für alle gesorgt, die nicht darauf verzichten wollen, ihr Directory in der gewohnten Weise zu sehen. Ein Druck auf <HELP> bringt das Directory in der gewohnten Weise auf den Bildschirm. Zusätzlich können Sie unter folgenden Möglichkeiten wählen:

<F1> Anzeige aller PRG-Files
<F2> Anzeige aller USR-Files
<F3> Anzeige aller SEQ-Files
<F4> Anzeige aller REL-Files
<F7> Laufwerk umschalten
<F8> Ausgabe zwischen Drucker und Monitor umschalten
<ESC> zurück ins Hauptmenü

Jede andere Taste (einschließlich <HELP>) bringt ein vollständiges Directory auf das aktuelle Ausgabe-Gerät.

<ESC> Programm beenden

Damit wird das Programm zwar beendet, aber nicht aus dem Speicher entfernt, es bleibt resident und kann jederzeit mit SYS 4864 wieder gestartet werden.

(Johann Waldhäusl/so)

```
16c8 : a9 00 8d 00 ff ad 06 d5  06
16d0 : 09 07 8d 06 d5 8e 00 ff  53
16d8 : a2 80 86 f7 a9 00 20 90  97
16e0 : ff a9 17 a2 69 85 2d 86  52
16e8 : 2e 18 69 02 8d 10 12 8a  74
16f0 : 69 00 8d 11 12 a9 01 a2  97
16f8 : 41 85 2f 86 30 85 31 86  99
1700 : 32 85 33 86 34 a9 ff a2  68
1708 : e9 85 35 86 36 85 39 86  53
1710 : 3a a0 00 a9 01 99 00 10  cd
1718 : c8 c0 0a d0 f8 a2 81 8a  9d
1720 : 99 00 10 c8 e8 e0 8b d0  3c
1728 : f6 20 f9 24 20 5b 53 20  9c
1730 : a6 28 20 f1 24 ad 9f 13  85
1738 : 8d 1c 01 4c d4 34 a9 00  33
1740 : aa 4c 16 2f a9 ff a2 00  a1
1748 : 4c 16 2f a9 00 a2 ff 4c  4e
1750 : 16 2f a9 ff aa 4c 16 2f  2c
1758 : a9 00 a2 ea 85 fc 86 fd  5e
1760 : a9 00 8d 98 13 8d 99 13  aa
1768 : a0 04 20 e6 24 b1 fc 18  e3
1770 : 6d 98 13 8d 98 13 90 04  0c
1778 : ee 99 13 18 c8 c8 c8 c8  82
1780 : c0 48 f0 f8 c0 8d 90 e5  46
1788 : ad 93 13 f0 14 a0 dd 18  d0
1790 : b1 fc 6d 98 13 8d 98 13  54
1798 : 90 03 ee 99 13 18 c8 d0  4f
17a0 : ef 20 f1 24 60 ad 74 25  30
17a8 : 48 a9 0a 8d 74 25 a0 00  ec
17b0 : ae 99 13 ad 98 13 20 24  90
17b8 : 25 68 8d 74 25 60 a9 00  ff
17c0 : 8d 9a 13 8d 9b 13 8d 38  0a
17c8 : 16 ee 38 16 20 39 16 a5  96
17d0 : fc d0 05 a5 fd d0 01 60  55
17d8 : a0 1c 20 04 25 a8 8a 18  01
17e0 : 6d 9a 13 8d 9a 13 98 6d  90
17e8 : 9b 13 8d 9b 13 4c c9 17  cd
17f0 : 20 be 17 ad 74 25 48 a9  d0
17f8 : 0a 8d 74 25 a0 00 ad 9a  81
1800 : 13 ae 9b 13 20 24 25 68  3c
1808 : 8d 74 25 60 ad 74 25 48  c8
1810 : a9 0a 8d 74 25 18 ad 9a  af
1818 : 13 6d 98 13 48 ad 9b 13  f1
1820 : 6d 99 13 aa 68 18 6d 9c  aa
1828 : 13 48 8a 69 00 aa 68 a0  67
1830 : 00 20 24 25 68 8d 74 25  fd
1838 : 60 ad 90 13 c9 03 d0 03  f3
1840 : 20 62 13 60 ad 90 13 c9  a1
1848 : 03 d0 03 20 68 13 60 20  59
1850 : 36 24 20 5f 13 20 6e 13  9e
1858 : 20 b9 25 20 7d ff 49 4e  3c
1860 : 53 45 52 54 20 44 49 53  65
1868 : 4b 20 54 4f 20 42 45 20  2c
1870 : 48 41 4e 44 4c 45 44 20  b5
1878 : 41 4e 44 20 48 49 54 20  56
1880 : 20 20 12 0f 20 20 53 50  08
1888 : 41 43 45 20 20 92 8f 00  95
1890 : 20 68 13 20 e4 ff c9 1b  59
1898 : f0 04 c9 20 d0 f5 60 a2  84
18a0 : 11 a0 0a 20 c3 25 20 6e  4b
18a8 : 13 20 7d ff 50 4c 45 41  2a
18b0 : 53 45 20 53 45 4c 45 43  6b
18b8 : 54 20 3a 20 00 a9 17 a2  9e
18c0 : 1a 60 20 4f 18 c9 1b d0  da
18c8 : 03 4c bb 19 20 5f 13 20  8d
18d0 : c2 24 20 44 17 ad 81 33  80
18d8 : 10 03 4c bb 19 20 2c 15  62
18e0 : 20 5f 13 20 72 2b 20 7a  6f
18e8 : 13 20 aa 2a a9 00 8d 38  3c
18f0 : 16 ad 38 16 8d d8 2c a9  51
18f8 : 06 a2 00 20 2e ca 20 5f  cc
1900 : 13 a9 03 a2 10 a0 03 20  4f
1908 : d7 2b 20 e4 ff c9 2b f0  f6
1910 : 03 4c 2d 19 20 b3 25 ae  39
1918 : 38 16 e0 90 f0 05 ad b0  f5
1920 : 2c f0 ce a9 00 8d 38 16  27
1928 : f0 c7 4c b1 19 c9 81 f0  0d
1930 : 03 4c 4a 19 20 9f 18 a0  b0
1938 : 0c 20 6f 29 20 1e 2d 20  3d
1940 : 8e 1a 20 b1 2c a9 00 4c  c2
1948 : b1 19 c9 82 f0 03 4c 67  70
1950 : 19 20 9f 18 a0 04 20 6f  ee
1958 : 29 20 1e 2d 20 ff 1b 20  6d
1960 : b1 2c a9 00 4c b1 19 c9  dc
1968 : 83 f0 03 4c 79 19 20 92  b4
1970 : 1c 20 b1 2c a9 00 4c b1  bd
1978 : 19 c9 86 f0 03 4c 8f 19  38
1980 : a9 1b a2 1e 20 ff 22 20  ee
1988 : b1 2c a9 00 4c b1 19 c9  04
1990 : 85 f0 03 4c a5 19 a9 02  a6
1998 : a2 13 20 ff 22 20 b1 2c  0e
19a0 : a9 00 4c b1 19 c9 55 f0  aa
19a8 : 03 4c b1 19 20 84 33 a9  a7
19b0 : 1b c9 1b d0 03 4c bb 19  44
19b8 : 4c 0a 19 20 5f 13 a9 00  89
19c0 : 60 20 e6 24 20 77 1a a0  d6
19c8 : 00 b1 fc aa c8 b1 fc c8  d5
19d0 : c8 91 fc 88 8a 91 fc ce  78
19d8 : 0e 1a ad 0e 1a cd 0f 1a  a1
19e0 : b0 e2 20 f1 24 60 80 12  b3
19e8 : 00 2d 2d 2d 2d 2d 2d 2d  bb
19f0 : 2d 2d 2d 2d 2d 2d 2d 2d  f0
19f8 : 2d 00 00 00 00 00 00 00  26
1a00 : 00 00 00 00 00 00 00 00  01
1a08 : 00 00 00 00 00 00 00 00  09
1a10 : 00 00 00 00 00 00 00 00  11
1a18 : 00 00 00 00 00 00 00 00  19
1a20 : 00 00 00 00 00 00 00 00  21
1a28 : 00 00 00 00 00 00 00 00  29
1a30 : 00 00 00 00 00 00 00 00  31
1a38 : 00 00 00 01 20 e6 24 20  63
1a40 : 77 1a a0 00 b1 fc 8d 10  46
1a48 : 1a c8 b1 fc 8d 11 1a 4c  35
1a50 : 55 1a 20 77 1a a0 02 b1  bb
1a58 : fc 88 88 91 fc c8 c8 c8  b7
1a60 : b1 fc 88 88 91 fc ee 0e  9b
1a68 : 1a ad 0e 1a cd 0f 1a f0  bf
1a70 : e1 90 df 20 f1 24 60 a2  9c
```

Listing 1. »Power 128/v1.1« bitte mit dem MSE (Seite 158) eingeben

```
1a78 : 00 a0 fd 86 fc 84 fd 8d  1f
1a80 : 0e 1a 0a 90 02 e6 fd 85  8a
1a88 : fc 60 20 89 13 60 ad cd  74
1a90 : 2d d0 0c ad cc 2d 8d cd  e6
1a98 : 2d ad cb 2d 8d cc 2d 38  99
1aa0 : ad cc 2d ed cb 2d c9 15  34
1aa8 : b0 e0 8d 12 1a 6d cd 2d  0d
1ab0 : ae 3b 1a f0 04 38 ed 12  7e
1ab8 : 1a cd 28 15 b0 cc ad cb  25
1ac0 : 2d 20 77 1a 20 e6 24 a0  2a
1ac8 : 00 ce 0e 1a ee 0e 1a b1  21
1ad0 : fc 99 13 1a c8 b1 fc 99  e2
1ad8 : 13 1a c8 ad 0e 1a cd cc  63
1ae0 : 2d 90 e9 20 f1 24 ad cd  66
1ae8 : 2d cd cb 2d 90 38 cd cc  30
1af0 : 2d b0 03 4c 8a 1a ae 3b  6a
1af8 : 1a f0 0c 8d 0f 1a 38 ed  be
1b00 : 12 1a 8d cd 2d 10 07 18  dc
1b08 : 6d 12 1a 8d 0f 1a ad cc  c9
1b10 : 2d 20 3c 1a ce 0f 1a ce  0b
1b18 : cc 2d ad cc 2d cd cb 2d  4a
1b20 : 10 ef 20 47 1b 60 ae cd  24
1b28 : 2d 8e 0f 1a ce cb 2d ad  ff
1b30 : cb 2d 20 c1 19 ee 0f 1a  4c
1b38 : ee cb 2d ad cb 2d cd cc  04
1b40 : 2d d0 ef 20 47 1b 60 ad  ff
1b48 : cd 2d 20 e6 24 20 77 1a  e6
1b50 : a0 00 b9 13 1a 91 fc c8  75
1b58 : b9 13 1a 91 fc c8 ce 12  c9
1b60 : 1a 10 ef 20 f1 24 60 a9  97
1b68 : 6a a2 5d a0 02 20 a3 13  65
1b70 : 20 c2 18 48 a9 29 a2 5c  28
1b78 : a0 02 20 a3 13 68 60 8e  a9
1b80 : 10 1a 20 9f 18 a0 08 20  80
1b88 : 6f 29 20 1e 2d ad cb 2d  21
1b90 : f0 2f 8d 38 16 20 39 16  f6
1b98 : 20 e6 24 a0 1c ae 10 1a  f4
1ba0 : e0 85 d0 04 a9 00 f0 02  5a
1ba8 : a9 ff 91 fc c8 91 fc 20  a2
1bb0 : f1 24 ad cc 2d f0 0a cd  d6
1bb8 : cb 2d f0 05 ee cb 2d d0  9a
1bc0 : cc 20 b1 2c a9 00 60 ad  06
1bc8 : cb 2d f0 30 8d 38 16 8a  74
1bd0 : 48 20 39 16 68 aa 20 e6  64
1bd8 : 24 a0 00 b1 fc e0 83 f0  49
1be0 : 05 29 bf 4c e8 1b 09 40  ff
1be8 : 91 fc 20 f1 24 ad cc 2d  7b
1bf0 : f0 0a cd cb 2d f0 05 ee  1e
1bf8 : cb 2d d0 cb a9 00 60 ad  7f
1c00 : 28 15 8d 0f 1a ad cb 2d  91
1c08 : d0 03 4c 91 1c 20 3c 1a  87
1c10 : ce 28 15 20 36 24 20 b1  a4
1c18 : 2c 20 6e 13 20 7d ff 0d  5a
1c20 : 0d 20 20 20 20 20 20 20  0d
1c28 : 48 49 54 20 20 3c 20 45  1d
1c30 : 53 43 20 3e 20 20 54 4f  e8
1c38 : 20 52 45 43 4f 56 45 52  9c
1c40 : 20 4f 52 20 20 3c 20 52  aa
1c48 : 45 54 55 52 4e 20 3e 20  76
1c50 : 20 54 4f 20 41 43 43 45  38
1c58 : 50 54 00 20 e4 ff c9 1b  82
1c60 : f0 04 c9 0d d0 f5 c9 1b  80
1c68 : d0 27 ad cb 2d 8d 0f 1a  60
1c70 : ad 28 15 8d 0e 1a 20 c1  de
1c78 : 19 ad 10 1a aa ad 11 1a  40
1c80 : a0 01 20 e6 24 91 fc 8a  5e
1c88 : 88 91 fc 20 f1 24 ee 28  68
1c90 : 15 60 ad 28 15 c9 91 90  4d
1c98 : 0d ee 28 15 ce 28 15 a9  9f
1ca0 : 07 20 d2 ff d0 30 20 9f  ba
1ca8 : 18 a0 04 20 6f 29 ee 28  62
1cb0 : 15 20 1e 2d ad cb 2d f0  d2
1cb8 : e3 8d 0f 1a ad 28 15 8d  f4
1cc0 : 0e 1a 20 c1 19 20 e6 24  92
1cc8 : a0 00 a9 e6 91 fc c8 a9  27
1cd0 : 19 91 fc 20 f1 24 60 20  f7
1cd8 : 4f 18 c9 1b d0 03 4c 71  42
1ce0 : 1e 20 5f 13 20 c2 24 20  32
1ce8 : 44 17 ad 81 33 10 03 4c  ac
1cf0 : 71 1e 20 2c 15 20 5f 13  f4
1cf8 : 20 72 2b 20 7a 13 20 aa  36
1d00 : 2a ad 28 15 8d 4a 1f a9  a9
1d08 : 00 8d 38 16 ad 38 16 8d  b0
1d10 : d8 2c a9 06 a2 00 20 2e  30
1d18 : ca 20 5f 13 a9 03 a2 10  8a
1d20 : a0 03 20 d7 2b 20 e4 ff  8c
1d28 : c9 2b f0 03 4c 48 1d 20  df
1d30 : b3 25 ae 38 16 e0 90 f0  b5
1d38 : 05 ad b0 2c f0 ce a9 00  f2
1d40 : 8d 38 16 f0 c7 4c 67 1e  46
1d48 : c9 81 f0 03 4c 65 1d 20  13
1d50 : 9f 18 a0 0c 20 6f 29 20  07
1d58 : 1e 2d 20 8e 1a 20 b1 2c  a8
1d60 : a9 00 4c 67 1e c9 82 f0  25
1d68 : 03 4c 82 1d 20 9f 18 a0  76
1d70 : 04 20 6f 29 20 1e 2d 20  6d
1d78 : ff 1b 20 b1 2c a9 00 4c  ec
1d80 : 67 1e c9 83 f0 03 4c 94  5b
1d88 : 1d 20 92 1c 20 b1 2c a9  71
1d90 : 00 4c 67 1e c9 85 f0 03  e6
1d98 : 4c ef 1d 8d 9d 1e 20 5f  df
1da0 : 13 a2 06 a0 0a 20 c3 25  95
1da8 : ad 51 20 f0 1a a9 2d 20  08
1db0 : e8 29 a9 3c a2 09 a0 00  14
1db8 : 20 61 29 a9 0f a2 0f a0  8c
1dc0 : 2d 20 a2 29 4c de 1d a9  4f
1dc8 : 3c 20 e8 29 a9 4b a2 09  05
1dd0 : a0 00 20 61 29 a9 0f a2  06
1dd8 : 0f a0 3c 20 a2 29 18 ad  7a
1de0 : 9e 13 69 0f 8d 9e 13 20  9e
1de8 : b1 2c a9 00 4c 67 1e c9  26
1df0 : 88 f0 03 4c 22 1e ad 51  a7
1df8 : 20 f0 06 20 4c 1f 4c 15  2f
1e00 : 1e 20 9f 18 a0 04 20 6f  a3
1e08 : 29 20 1e 2d ae cb 2d f0  4e
1e10 : 09 e8 8e 28 15 a9 00 8d  f0
1e18 : 38 16 20 b1 2c a9 00 4c  42
1e20 : 67 1e c9 87 f0 03 4c 43  d8
1e28 : 1e 20 5f 13 20 62 13 a2  37
1e30 : 01 8e 91 13 ca 8e 38 16  6d
1e38 : 20 c0 1e a9 00 8d 91 13  4e
1e40 : 4c 67 1e c9 50 f0 03 4c  32
1e48 : 67 1e 20 e8 14 a2 01 20  7e
1e50 : c9 ff a9 00 cd 91 13 f0  1b
1e58 : 0c 8d 38 16 20 c0 1e 20  bc
1e60 : 7a 13 20 e8 14 a9 1b c9  18
1e68 : 1b d0 03 4c 71 1e 4c 25  b9
1e70 : 1d 20 5f 13 a9 00 60 a9  47
1e78 : ec a2 5f a0 02 20 a3 13  77
1e80 : ad 51 20 f0 09 a9 2d a2  d4
1e88 : 61 a0 04 20 a3 13 20 d7  41
1e90 : 1c 48 a9 ab a2 5e a0 02  54
1e98 : 20 a3 13 68 60 00 ad 28  69
1ea0 : 15 c9 20 b0 03 a2 0a 60  e6
1ea8 : c9 3e b0 03 a2 14 60 c9  fd
1eb0 : 5c b0 03 a2 1e 60 c9 7a  7a
1eb8 : b0 03 a2 28 60 a2 32 60  3c
1ec0 : a9 01 8d 20 27 ad 51 20  b7
1ec8 : f0 03 4c 53 20 ad 28 15  f2
1ed0 : 48 ad 4a 1f 8d 28 15 ad  2f
1ed8 : 90 13 c9 03 f0 06 20 09  96
1ee0 : 22 20 57 22 a2 4f a9 3d  f2
1ee8 : 20 d2 ff ca d0 f8 20 7a  15
1ef0 : 13 20 83 21 20 7a 13 68  0b
1ef8 : 8d 28 15 ad 9d 1e f0 16  4f
1f00 : a2 4f a9 2d 20 d2 ff ca  88
1f08 : d0 f8 20 7a 13 ad 9e 13  eb
1f10 : 20 66 29 20 7a 13 a2 4f  1b
1f18 : a9 2d 20 d2 ff ca d0 f8  46
1f20 : 20 7a 13 20 7a 13 20 9e  44
1f28 : 1e ad 4b 1f a0 03 20 d7  26
1f30 : 2b 20 7a 13 a2 4f a9 3d  32
1f38 : 20 d2 ff ca d0 f8 20 7a  65
1f40 : 13 20 7a 13 a9 00 8d 20  75
1f48 : 27 60 00 03 8d 4b 20 20  f4
1f50 : 5f 13 a2 06 a0 0a 20 c3  05
1f58 : 25 20 cc ff 20 6e 13 20  c2
1f60 : 7d ff 50 4c 45 41 53 45  b1
1f68 : 20 45 4e 54 45 52 20 4c  49
1f70 : 41 42 45 4c 2d 48 45 41  5a
1f78 : 44 45 52 20 20 28 20 00  bb
1f80 : ad 4c 20 20 fb f9 48 8a  25
1f88 : 20 d2 ff 68 20 d2 ff 20  f7
1f90 : 7d ff 20 43 48 41 52 2e  b2
1f98 : 20 4d 41 58 2e 20 29 00  43
1fa0 : a2 08 a0 00 20 c3 25 a9  76
1fa8 : 21 20 d2 ff ae 4d 20 ca  f9
1fb0 : a9 2d 20 d2 ff ca d0 f8  de
1fb8 : a9 21 20 d2 ff ae 4e 20  43
1fc0 : ca ca a9 2d 20 d2 ff ca  2e
1fc8 : d0 f8 a9 21 20 d2 ff 20  7c
1fd0 : 68 13 a2 09 a0 00 20 c3  9e
1fd8 : 25 ae 4c 20 a9 1d a0 20  b2
1fe0 : 20 88 28 a9 0f a2 00 ac  e3
1fe8 : 4c 20 20 6f 29 20 b3 25  e7
1ff0 : a9 1d a0 20 20 b8 28 20  fd
1ff8 : e2 ca 60 a9 12 a2 00 8d  de
2000 : 7f 33 8e 80 33 a9 00 a2  92
2008 : ea 85 fe 86 ff 20 3e 2f  9e
2010 : 20 30 30 20 8a 2f 20 5f  ba
2018 : 13 20 aa 2a 60 20 20 20  f3
2020 : 20 20 20 20 20 20 20 20  20
2028 : 20 20 20 20 20 20 20 20  28
2030 : 20 20 20 20 20 20 20 20  30
2038 : 20 20 20 20 20 20 20 20  38
2040 : 20 20 20 20 20 20 20 20  40
2048 : 20 20 20 00 1b 15 06 0d  0d
2050 : 03 00 03 ad 28 15 48 ad  72
2058 : 4a 1f 8d 28 15 ad 90 13  c1
2060 : c9 03 f0 09 20 75 22 20  7f
2068 : 5d 22 20 69 22 ad 4b 20  09
2070 : d0 03 4c cb 20 ad 90 13  26
2078 : c9 03 f0 06 20 0f 22 20  03
2080 : 27 22 ae 4d 20 a9 1d a0  13
2088 : 20 20 88 28 ad 90 13 c9  1f
2090 : 03 f0 12 20 09 22 20 2d  10
2098 : 22 20 0f 22 20 27 22 20  d7
20a0 : 21 22 20 33 22 ae 4e 20  52
20a8 : 18 ad 4d 20 69 1d 48 a9  e2
20b0 : 20 69 00 a8 68 20 88 28  94
20b8 : ad 90 13 c9 03 f0 09 20  c7
20c0 : 2d 22 20 39 22 20 6f 22  53
20c8 : 20 7a 13 ad 90 13 c9 03  6f
20d0 : f0 09 20 09 22 20 27 22  72
20d8 : 20 15 22 20 83 21 ad 90  29
20e0 : 13 c9 03 f0 09 20 2d 22  41
20e8 : 20 1b 22 20 6f 22 68 8d  e7
20f0 : 28 15 ad 9d 1e f0 1c 20  dc
20f8 : 7a 13 ad 90 13 c9 03 f0  e7
2100 : 03 20 15 22 a9 3c 20 66  67
2108 : 29 ad 90 13 c9 03 f0 03  0d
2110 : 20 1b 22 20 7a 13 ad 90  63
2118 : 13 c9 03 f0 0f 20 09 22  49
2120 : 20 27 22 20 15 22 20 3f  c2
2128 : 22 20 7b 22 ae 4f 20 ad  bf
2130 : 4b 20 f0 02 ca ca ad 9d  fd
2138 : 1e f0 02 ca ca ad 90 13  2b
2140 : c9 03 f0 14 8a 48 ca ca  f5
2148 : 8a 18 6d 52 20 8d fc 22  2b
2150 : 20 81 22 20 5d 22 68 aa  9b
2158 : ac 08 22 ad 50 20 20 d7  7d
2160 : 2b ad 90 13 c9 03 f0 12  85
```

```
2168 : 20 2d 22 ad 9d 13 20 d2  f6
2170 : ff 20 4b 22 20 1b 22 20  3a
2178 : 75 22 a9 00 8d 20 27 8d  fa
2180 : 38 16 60 20 c8 25 a2 03  26
2188 : a9 20 20 d2 ff ca d0 f8  2f
2190 : 20 d8 25 ad 51 20 d0 0a  89
2198 : a2 06 a9 20 20 d2 ff ca  da
21a0 : d0 f8 a2 02 a9 20 20 d2  97
21a8 : ff ca d0 f8 20 02 26 a2  4f
21b0 : 03 a9 20 20 d2 ff ca d0  8e
21b8 : f8 20 e2 26 a2 03 a9 20  67
21c0 : 20 d2 ff ca d0 f8 a2 5d  bd
21c8 : bd 97 3c 20 d2 ff e8 e0  f6
21d0 : 62 d0 f5 ae 28 15 ca a0  85
21d8 : 00 8a a2 00 20 24 25 20  be
21e0 : e8 25 20 58 17 20 a5 17  a5
21e8 : ad 51 20 d0 0d a2 2c bd  72
21f0 : 97 3c 20 d2 ff e8 e0 33  39
21f8 : d0 f5 a2 33 bd 97 3c 20  9c
2200 : d2 ff e8 e0 38 d0 f5 60  cb
2208 : 04 a2 00 a0 02 d0 78 a2  3f
2210 : 05 a0 02 d0 72 a2 0a a0  a6
2218 : 02 d0 6c a2 0f a0 01 d0  8d
2220 : 66 a2 14 a0 01 d0 60 a2  4e
2228 : 19 a0 02 d0 5a a2 1e a0  a0
2230 : 02 d0 54 a2 23 a0 02 d0  e5
2238 : 4e a2 28 a0 02 d0 48 a2  03
2240 : 2d a0 03 d0 42 a2 32 a0  db
2248 : 03 d0 3c a2 37 a0 02 d0  39
2250 : 36 a2 3c a0 03 d0 30 a2  b7
2258 : 41 a0 03 d0 2a a2 46 a0  d6
2260 : 02 d0 24 a2 4b a0 03 d0  8f
2268 : 1e a2 50 a0 03 d0 18 a2  5c
2270 : 55 a0 03 d0 12 a2 5a a0  d1
2278 : 02 d0 0c a2 5f a0 03 d0  e2
2280 : 06 a2 64 a0 03 d0 00 c0  3d
2288 : 00 f0 0a bd 96 22 20 d2  db
2290 : ff e8 88 d0 f6 60 1b 4d  b9
2298 : 00 00 00 1b 50 00 00 00  01
22a0 : 1b 0f 00 00 00 12 00 00  d4
22a8 : 00 00 0e 00 00 00 00 1b  62
22b0 : 45 00 00 00 1b 46 00 00  d9
22b8 : 00 1b 47 00 00 00 1b 48  15
22c0 : 00 00 00 1b 53 01 00 00  61
22c8 : 1b 53 00 00 00 1b 54 00  b7
22d0 : 00 00 1b 4a 14 00 00 1b  58
22d8 : 6c 0a 00 00 1b 4f 00 00  76
22e0 : 00 1b 4e 05 00 00 1b 2d  69
22e8 : 01 00 00 1b 2d 00 00 00  20
22f0 : 1b 40 00 00 00 1b 41 06  15
22f8 : 00 00 1b 43 00 00 00 8d  43
2300 : 96 23 8e 97 23 a9 01 8d  5d
2308 : 91 23 ee 91 23 ad 91 23  45
2310 : cd 28 15 90 01 60 ad 91  36
2318 : 23 8d 92 23 ad 96 23 8d  42
2320 : 93 23 ee 93 23 ce 92 23  ac
2328 : f0 40 ad 91 23 8d 38 16  82
2330 : 20 39 16 ac 93 23 20 e6  a9
2338 : 24 b1 fc 20 f1 24 8d 94  18
2340 : 23 ad 92 23 8d 38 16 20  76
2348 : 39 16 ac 93 23 20 e6 24  41
2350 : b1 fc 20 f1 24 8d 95 23  11
2358 : c9 a0 f0 05 cd 94 23 b0  bd
2360 : 21 ae 92 23 e8 ec 91 23  64
2368 : f0 a0 ad 91 23 8d cb 2d  6e
2370 : ae 92 23 e8 8e cc 2d a9  a5
2378 : 00 8d cd 2d 20 8e 1a 4c  cf
2380 : 0a 23 d0 98 ad 93 23 cd  03
2388 : 97 23 b0 de ee 92 23 d0  6a
2390 : 91 00 00 00 00 00 00 00  22
2398 : a9 00 a2 ea 85 fe 86 ff  b2
23a0 : a2 00 bd 17 29 20 e6 24  0c
23a8 : 91 fe 20 f1 24 ec 14 29  4b
23b0 : b0 04 e8 c8 d0 ec 60 a0  ed
23b8 : 00 a2 ea 84 fe 86 ff a0  ba
23c0 : ff 20 e6 24 91 fe 20 f1  83
23c8 : 24 88 c0 04 90 0c c0 90  6f
23d0 : 90 ef c0 dd b0 eb a0 8f  50
23d8 : d0 e7 60 a2 3c a9 00 9d  55
23e0 : 17 29 ca 10 fa 60 a2 00  7e
23e8 : bd 17 29 f0 03 e8 d0 f8  46
23f0 : ca 8e 35 24 20 9f 18 a0  74
23f8 : 08 20 6f 29 20 1e 2d ad  14
2400 : cb 2d f0 2d 8d 38 16 20  77
2408 : 39 16 a0 12 ae 35 24 20  1c
2410 : e6 24 bd 17 29 91 fc 88  7f
2418 : ca 10 f7 a9 a0 91 fc 20  e8
2420 : f1 24 ad cc 2d f0 0a cd  46
2428 : cb 2d f0 05 ee cb 2d d0  0a
2430 : ce 20 b1 2c 60 00 a9 02  b1
2438 : a2 00 20 2e ca a9 17 a2  44
2440 : 4f 20 1d ca 60 a9 00 20  d4
2448 : c3 ff a9 0d 20 c3 ff 20  77
2450 : cc ff 20 89 13 a9 15 a2  6d
2458 : 0f 20 2e ca a9 18 a2 41  c5
2460 : 20 1d ca a9 0f 20 d2 ff  34
2468 : 20 5f 13 20 6b 13 20 b3  38
2470 : 25 a9 ff 8d 81 33 20 7d  49
2478 : ff 20 53 4f 52 52 59 2c  bc
2480 : 20 00 60 20 7d ff 20 20  55
2488 : 20 20 44 45 42 55 47 20  9e
2490 : 53 59 53 54 45 4d 20 41  b1
2498 : 4e 44 20 48 49 54 20 20  11
24a0 : 3c 53 50 41 43 45 3e 00  19
24a8 : 20 e4 ff c9 1b f0 04 c9  50
24b0 : 20 d0 f5 a9 8f 20 d2 ff  30
24b8 : 20 68 13 20 5f 13 20 36  51
24c0 : 24 60 20 36 24 20 5f 13  ca
24c8 : a2 00 a0 00 20 c3 25 20  87
24d0 : 71 13 20 7d ff 49 27 4d  04
24d8 : 20 42 55 53 59 20 2e 2e  85
24e0 : 2e 00 20 68 13 60 48 a9  cc
24e8 : 7f 8d 00 ff 68 60 20 cc  d1
24f0 : ff 48 a9 02 8d 00 ff 68  68
24f8 : 60 48 ad 00 ff 09 02 8d  53
2500 : 00 ff 68 60 20 e6 24 b1  53
2508 : fc aa c8 b1 fc 20 f1 24  a2
2510 : 60 85 60 86 61 84 62 20  20
2518 : 07 ba a9 00 a2 08 a0 03  da
2520 : 20 5d ba 60 85 60 86 61  e2
2528 : 84 62 20 07 ba a9 00 a2  05
2530 : 08 a0 03 8d b4 0a 8c b6  36
2538 : 0a ac b6 0a a9 00 0e a3  a1
2540 : 0a 2e a2 0a 2e a1 0a 2e  c0
2548 : a0 0a 2a 88 10 f0 a8 d0  56
2550 : 09 e0 01 f0 05 ac b4 0a  c4
2558 : f0 11 ec 74 25 10 08 ee  6b
2560 : b4 0a 09 30 20 d2 ff ca  90
2568 : d0 cf 60 ec 74 25 10 f7  76
2570 : a9 20 d0 f0 05 a9 01 2c  75
2578 : a9 00 85 80 20 36 24 20  17
2580 : 68 13 20 5f 13 a2 00 a0  ed
2588 : 16 20 c3 25 a2 16 a9 03  cb
2590 : 20 2e ca a2 36 a9 16 20  18
2598 : 1d ca a9 00 85 81 20 8a  7f
25a0 : a0 20 f1 24 20 36 24 20  d6
25a8 : 3e 2f 20 8a 2f a9 00 60  d8
25b0 : 38 b0 11 a2 00 a0 00 f0  c0
25b8 : 0a 20 36 24 20 5f 13 a2  73
25c0 : 0a a0 0a 18 20 f0 ff 60  ea
25c8 : a9 00 a2 ea 85 fc 86 fd  ce
25d0 : a0 8f a9 10 20 6c 16 60  23
25d8 : a9 00 a2 ea 85 fc 86 fd  de
25e0 : a0 a1 a9 05 20 6c 16 60  da
25e8 : a9 00 a2 ea 85 fc 86 fd  ee
25f0 : a0 02 20 04 25 8d 93 13  4d
25f8 : 29 80 d0 05 a0 ab 20 04  26
2600 : 25 60 20 e8 25 f0 34 c9  b9
2608 : a0 f0 18 a2 05 a9 20 20  d9
2610 : d2 ff ca d0 f8 a2 00 bd  cf
2618 : 49 26 20 d2 ff e8 e0 04  a9
2620 : d0 f5 60 a2 05 a9 20 20  b6
2628 : d2 ff ca d0 f8 a2 04 bd  f7
2630 : 49 26 20 d2 ff e8 e0 08  ca
2638 : d0 f5 60 a2 04 bd 49 26  0f
2640 : 20 d2 ff e8 e0 0d d0 f5  8c
2648 : 60 31 35 37 31 31 35 37  55
2650 : 30 2f 31 35 34 31 a9 01  80
2658 : a2 00 a0 30 8c 24 15 8c  80
2660 : 25 15 8c 26 15 20 9e 26  11
2668 : ad 95 13 c9 43 d0 26 ad  8d
2670 : 96 13 c9 42 d0 1f ad 97  36
2678 : 13 c9 4d d0 18 a9 36 8d  a0
2680 : 26 15 a9 00 20 3e 2f 20  8c
2688 : ae 26 ae 95 13 e8 8e 9c  94
2690 : 13 8a 4c 9a 26 a9 00 8d  1a
2698 : 9c 13 8d 94 13 60 8d 7f  1d
26a0 : 33 8e 80 33 a9 00 20 3e  38
26a8 : 2f a9 31 20 e2 2f 20 0c  3c
26b0 : 15 a2 0d 20 c6 ff a0 03  53
26b8 : a2 00 20 cf ff 9d 95 13  c6
26c0 : e8 88 d0 f6 20 8a 2f 60  d3
26c8 : a0 00 a2 00 ad 9c 13 20  5d
26d0 : 24 25 60 a9 00 a2 ea 85  a0
26d8 : fc 86 fd a0 02 20 04 25  26
26e0 : 8a 60 a9 52 20 d2 ff a9  3b
26e8 : 2f 20 d2 ff 20 d3 26 c9  a9
26f0 : 41 f0 06 a9 4f 20 d2 ff  a1
26f8 : 60 a9 57 20 d2 ff 60 a0  f7
2700 : 00 20 04 25 8a 29 07 8d  df
2708 : 1f 27 0a 0a aa a0 03 bd  b6
2710 : 21 27 20 d2 ff ad 20 27  63
2718 : d0 04 e8 88 d0 f1 60 00  54
2720 : 00 44 45 4c 20 53 45 51  72
2728 : 20 50 52 47 20 55 53 52  8c
2730 : 20 52 45 4c a0 00 20 04  e7
2738 : 25 8a 29 80 f0 1f 8a 29  81
2740 : 40 f0 0d ad 6a 27 20 d2  f7
2748 : ff ad 6b 27 20 d2 ff 60  37
2750 : ad 6c 27 20 d2 ff ad 6d  c0
2758 : 27 20 d2 ff 60 ad 6e 27  bf
2760 : 20 d2 ff ad 6f 27 20 d2  f5
2768 : ff 60 52 4f 52 57 2a 2a  f3
2770 : a0 1c 20 04 25 a8 8a 48  f9
2778 : 98 a0 00 aa 68 8d 87 27  15
2780 : 8e 88 27 20 24 25 60 00  0d
2788 : 00 20 9d 27 a2 01 20 c9  2b
2790 : ff a0 00 ad 95 13 ae 96  6f
2798 : 13 20 24 25 60 ad 94 13  55
27a0 : f0 15 ad 80 33 d0 10 ad  ec
27a8 : 7f 33 c9 01 d0 09 a9 00  4f
27b0 : 8d 95 13 8d 96 13 60 a9  55
27b8 : 30 8d 24 15 8d 25 15 a9  04
27c0 : 32 8d 26 15 20 a4 26 60  66
27c8 : ad 95 13 d0 09 ad 96 13  9d
27d0 : d0 04 aa 4c f7 27 ad 87  55
27d8 : 27 ae 88 27 85 2a 86 2b  77
27e0 : a9 00 a2 01 85 72 86 73  3f
27e8 : 20 fd 27 8a 18 6d 95 13  8b
27f0 : 48 98 6d 96 13 aa 68 a0  1c
27f8 : 00 20 24 25 60 a9 10 85  55
2800 : 5f 18 a2 00 a0 00 8a 0a  5c
2808 : aa 98 2a a8 b0 16 06 72  57
2810 : 26 73 90 0b 18 8a 65 2a  35
2818 : aa 98 65 2b a8 b0 05 c6  7f
2820 : 5f d0 e3 60 a9 00 aa a8  83
2828 : 18 60 ad 7c 33 cd 9f 13  b1
2830 : f0 05 ad 9f 13 d0 03 ad  21
```

Listing 1. »Power 128/v.1.1« (Fortsetzung)

```
2838 : a0 13 8d 7c 33 8d 1c 01  67
2840 : a9 0e a0 0f ae 7c 33 20  d6
2848 : ba ff a9 00 20 bd ff 20  9c
2850 : c0 ff 18 a2 0e 20 c9 ff  73
2858 : 90 1c 20 45 24 ad 7c 33  af
2860 : 20 c7 31 ad 7c 33 cd 9f  3e
2868 : 13 d0 05 ad a0 13 d0 03  c6
2870 : ad 9f 13 8d 7c 33 20 cc  df
2878 : ff a9 0e 20 c3 ff ad 7c  bf
2880 : 33 8d 1c 01 20 36 24 60  a6
2888 : 8e 9e 13 8d 97 28 8c 98  fa
2890 : 28 a2 00 20 f9 24 bd 00  c5
2898 : 00 20 f1 24 20 d2 ff e8  14
28a0 : ec 9e 13 d0 ee 60 a9 02  57
28a8 : a0 00 a9 03 20 ba ff a9  3e
28b0 : 00 20 bd ff 20 c0 ff 60  f8
28b8 : 8d e8 28 8c e9 28 20 cc  4f
28c0 : ff 20 74 13 e8 20 cf ff  1e
28c8 : c9 0d d0 f8 20 77 13 20  b5
28d0 : b3 25 a2 02 20 c6 ff 86  44
28d8 : d6 a2 ff e8 a9 00 85 f4  b7
28e0 : 20 cf ff c9 0d f0 08 9d  d5
28e8 : 00 00 ec 9e 13 d0 ec ac  bc
28f0 : 14 2f f0 04 ca 8e 13 2f  24
28f8 : ac 59 53 f0 06 ca 8e 5a  e9
2900 : 53 10 0a 8e 14 29 e0 11  e0
2908 : b0 03 8e a9 2a 20 cc ff  e9
2910 : 20 e2 ca 60 00 4e 3a 20  fc
2918 : 20 20 20 20 20 20 20 20  18
2920 : 20 20 20 20 20 20 20 20  20
2928 : 20 20 20 20 20 20 20 20  28
2930 : 20 20 20 20 20 20 20 20  30
2938 : 20 20 20 20 20 20 20 20  38
2940 : 20 20 20 20 20 20 20 20  40
2948 : 20 20 20 20 20 20 20 20  48
2950 : 20 20 20 20 20 c3 25 a9  94
2958 : c3 a0 67 a2 0e 20 88 28  ee
2960 : 60 48 20 c3 25 68 aa a9  f8
2968 : d1 a0 67 20 88 28 60 8c  cc
2970 : 9e 13 a8 48 8a 48 98 20  59
2978 : 2e ca 68 18 6d 9e 13 aa  96
2980 : ca 68 20 1d ca 20 e5 ca  05
2988 : 20 b3 25 60 a0 0e 20 6f  b1
2990 : 29 20 57 29 20 b3 25 a9  4c
2998 : c3 a0 67 20 b8 28 20 e2  9c
29a0 : ca 60 20 6f 29 20 b3 25  3d
29a8 : a9 e0 a0 67 20 b8 28 20  7f
29b0 : e2 ca 60 20 cc ff 20 6e  3e
29b8 : 13 20 7d ff 50 4c 45 41  3a
29c0 : 53 45 20 45 4e 54 45 52  a8
29c8 : 20 44 41 54 45 20 20 28  0b
29d0 : 20 31 34 20 43 48 41 52  ba
29d8 : 2e 20 4d 41 58 2e 20 29  5c
29e0 : 20 3a 20 00 20 68 13 60  78
29e8 : 48 20 cc ff 20 6e 13 20  75
29f0 : 7d ff 50 4c 45 41 53 45  41
29f8 : 20 45 4e 54 45 52 20 52  e5
2a00 : 45 4d 41 52 4b 20 20 28  0d
2a08 : 20 00 68 20 fb f9 48 8a  0c
2a10 : 20 d2 ff 68 20 d2 ff 20  7f
2a18 : 7d ff 20 43 48 41 52 2e  3a
2a20 : 20 4d 41 58 2e 20 29 00  cb
2a28 : 20 68 13 60 20 cc ff 20  f6
2a30 : 6e 13 20 7d ff 50 4c 45  1e
2a38 : 41 53 45 20 45 4e 54 45  1b
2a40 : 52 20 46 49 4c 45 2d 4e  9d
2a48 : 41 4d 45 20 20 3a 00 a9  ac
2a50 : 07 a2 24 a0 10 20 6f 29  d8
2a58 : a2 00 ec a9 2a f0 0c bd  40
2a60 : 0d 41 20 d2 ff e8 ec a9  be
2a68 : 2a d0 f4 20 b3 25 20 68  f1
2a70 : 13 a9 0d a0 41 20 b8 28  f8
2a78 : 60 20 5f 13 a2 05 a0 0a  0b
2a80 : 20 c3 25 20 2c 2a 20 36  d0
```

```
2a88 : 24 20 5f 13 20 cc 24 a0  31
2a90 : 00 ae 7c 33 20 ba ff ad  a0
2a98 : a9 2a a2 0d a0 41 20 bd  b1
2aa0 : ff a9 01 a2 7f 20 68 ff  a3
2aa8 : 60 10 20 39 18 a2 00 bd  52
2ab0 : 97 3c 20 d2 ff e8 e0 07  a0
2ab8 : d0 f5 20 44 18 20 c8 25  03
2ac0 : 20 39 18 a2 08 bd 97 3c  1c
2ac8 : 20 d2 ff e8 e0 0c d0 f5  0c
2ad0 : 20 44 18 20 d8 25 20 39  c6
2ad8 : 18 a2 3c bd 97 3c 20 d2  8a
2ae0 : ff e8 e0 49 d0 f5 20 44  7a
2ae8 : 18 20 e2 26 ad df 47 d0  26
2af0 : 1f 20 39 18 a2 5b bd 97  9c
2af8 : 3c 20 d2 ff e8 e0 62 d0  ba
2b00 : f5 20 44 18 ae 28 15 ca  30
2b08 : a0 00 8a a2 00 20 24 25  7b
2b10 : 20 39 18 a2 2c bd 97 3c  af
2b18 : 20 d2 ff e8 e0 39 d0 f5  c5
2b20 : 20 44 18 20 e8 25 20 58  55
2b28 : 17 20 a5 17 20 7a 13 60  7e
2b30 : 20 44 18 ad 74 25 48 a9  13
2b38 : 04 8d 74 25 a2 00 a0 00  71
2b40 : ad 38 16 20 24 25 68 8d  bb
2b48 : 74 25 60 a0 00 20 04 25  d6
2b50 : 8a 48 29 07 aa bd 6d 2b  ce
2b58 : 20 d2 ff 68 48 29 80 d0  60
2b60 : 03 20 7d 13 68 29 40 f0  e8
2b68 : 03 20 74 13 60 1c 95 9e  75
2b70 : 1e 1f 20 b3 25 20 7d ff  e6
2b78 : 12 1c 20 20 20 20 44 45  43
2b80 : 4c 20 20 20 1d 1d 95 20  3a
2b88 : 20 20 53 45 51 20 20 20  0d
2b90 : 1d 1d 9e 20 20 20 50 52  d0
2b98 : 47 20 20 20 1d 1d 1e 20  6f
2ba0 : 20 20 55 53 52 20 20 20  77
2ba8 : 1d 1d 1f 20 20 20 52 45  f7
2bb0 : 4c 20 20 20 92 1d 1d 1d  d9
2bb8 : 1d 1d 1d 05 02 44 45 46  30
2bc0 : 45 43 54 49 56 45 82 1d  b9
2bc8 : 1d 12 50 52 4f 54 45 43  80
2bd0 : 54 45 44 92 0d 00 60 8d  98
2bd8 : ab 2c 8e ac 2c 8c ad 2c  09
2be0 : a2 00 8e b0 2c e8 8e ae  de
2be8 : 2c 8e af 2c ee 38 16 20  16
2bf0 : 39 16 ad 38 16 cd 28 15  41
2bf8 : 90 04 8d b0 2c 60 ae 91  a7
2c00 : 13 d0 0b 20 30 2b a9 20  85
2c08 : 20 d2 ff 20 4b 2b a9 10  6a
2c10 : a0 02 20 6c 16 ad 91 13  82
2c18 : d0 06 20 77 13 20 80 13  3d
2c20 : 20 70 27 ad 91 13 f0 15  97
2c28 : ad 51 20 d0 10 a9 2d 20  e3
2c30 : d2 ff 20 ff 26 a9 2d 20  ae
2c38 : d2 ff 20 34 27 ad ae 2c  8b
2c40 : cd ab 2c 90 03 4c 74 2c  bd
2c48 : ad 38 16 18 6d ac 2c cd  22
2c50 : 28 15 90 06 8d b0 2c 4c  8f
2c58 : 74 2c 8d 38 16 ee ae 2c  39
2c60 : ae ad 2c ad 91 13 f0 01  1d
2c68 : ca a9 20 20 d2 ff ca d0  0d
2c70 : f8 4c ef 2b ad af 2c cd  94
2c78 : ac 2c 90 01 60 ae ae 2c  0d
2c80 : ca ad 38 16 38 ed ac 2c  f0
2c88 : ca d0 f9 8d 38 16 a9 01  c7
2c90 : 8d ae 2c ad 91 13 d0 0a  3e
2c98 : a9 1b 20 d2 ff a9 51 20  04
2ca0 : d2 ff 20 7a 13 ee af 2c  89
2ca8 : 4c ec 2b 00 00 00 00 00  35
2cb0 : 00 20 36 24 ad d8 2c 8d  40
2cb8 : 38 16 a9 06 a2 00 20 2e  2d
2cc0 : ca 20 b3 25 a9 03 a2 10  89
2cc8 : a0 03 20 d7 2b a9 1b 20  9a
2cd0 : d2 ff a9 40 20 d2 ff 60  6e
```

```
2cd8 : 00 a2 00 86 16 86 17 c9  80
2ce0 : 2d f0 0c c9 a0 f0 08 c9  07
2ce8 : 30 90 04 c9 3a 90 01 60  88
2cf0 : e9 2f 85 09 a5 17 85 24  65
2cf8 : a5 16 0a 26 24 0a 26 24  63
2d00 : 65 16 85 16 a5 24 65 17  d4
2d08 : 85 17 06 16 26 17 a5 16  3b
2d10 : 65 09 85 16 90 02 e6 17  01
2d18 : 20 cf ff 4c df 2c 20 5f  48
2d20 : 13 20 cc ff 20 cf ff c9  8a
2d28 : 0d d0 f9 a9 1d 20 d2 ff  6f
2d30 : a9 a0 20 d2 ff a9 0d 20  4d
2d38 : d2 ff 20 b3 25 a9 00 8d  43
2d40 : ca 2d 8d cb 2d 8d cc 2d  4a
2d48 : 8d cd 2d a2 02 20 c6 ff  98
2d50 : 86 d6 20 cf ff c9 a0 f0  f6
2d58 : 62 c9 0d f0 5e c9 2d f0  cb
2d60 : 35 c9 30 b0 06 20 89 13  4a
2d68 : 4c 1e 2d c9 3a b0 f6 20  8d
2d70 : d9 2c c9 0d f0 45 a8 ad  ab
2d78 : c9 2d d0 0d a5 17 d0 e5  d0
2d80 : a5 16 f0 ce cd 28 15 b0  1a
2d88 : dc ae ca 2d 9d cb 2d ee  de
2d90 : ca 2d c0 2d d0 bc ae ca  0a
2d98 : 2d d0 0b e8 8a ca 9d cb  1a
2da0 : 2d ee ca 2d d0 ac e0 01  95
2da8 : d0 0d ac 28 15 88 98 ae  84
2db0 : ca 2d 9d cb 2d d0 9b a9  0d
2db8 : 01 d0 f4 20 cc ff 20 e2  75
2dc0 : ca a2 00 86 d6 8e c9 2d  10
2dc8 : 60 00 00 00 00 00 a0 00  ab
2dd0 : 84 fe a0 ea 84 ff c9 00  c8
2dd8 : f0 5a 8e 9c 13 a0 05 a9  ca
2de0 : ff 20 fb 2d e0 00 f0 61  28
2de8 : a9 ff c8 20 fb 2d e0 00  74
2df0 : f0 57 a9 0f c8 20 fb 2d  b0
2df8 : 4c 49 2e 0a f0 03 ca d0  aa
2e00 : fa 20 e6 24 31 fe 91 fe  98
2e08 : 20 f1 24 60 a9 00 38 2a  06
2e10 : c9 ff f0 03 ca d0 f7 20  c9
2e18 : e6 24 11 fe 91 fe 20 f1  aa
2e20 : 24 60 20 e6 24 b1 fe 20  65
2e28 : f1 24 a0 08 2a 90 01 e8  51
2e30 : 88 d0 f9 60 a0 05 20 0c  76
2e38 : 2e e0 00 f0 0c c8 20 0c  94
2e40 : 2e e0 00 f0 04 c8 20 0c  1c
2e48 : 2e a2 00 a0 05 20 22 2e  12
2e50 : a0 06 20 22 2e a0 07 20  84
2e58 : 22 2e a0 04 8a 20 e6 24  c8
2e60 : 91 fe 20 f1 24 a9 12 a2  d4
2e68 : 00 8d 7f 33 8e 82 33 8e  5c
2e70 : 7e 33 ad 83 33 48 8e 83  1a
2e78 : 33 20 55 33 68 8d 83 33  de
2e80 : 60 a9 00 a2 ea 85 fe e8  b2
2e88 : 86 ff a2 00 a0 00 bd 0c  d0
2e90 : 2f 20 e6 24 91 fe 20 f1  83
2e98 : 24 c8 e8 e0 07 90 ef a2  70
2ea0 : 00 ec 13 2f f0 14 bd 17  96
2ea8 : 29 20 e6 24 91 fe 20 f1  95
2eb0 : 24 e8 c8 ec 13 2f 90 ee  e3
2eb8 : f0 ec a9 00 20 e6 24 91  76
2ec0 : fe 20 f1 24 c8 a2 00 ec  4b
2ec8 : a9 2a f0 12 bd 0d 41 20  8e
2ed0 : e6 24 91 fe 20 f1 24 e8  00
2ed8 : c8 ec a9 2a 90 ee a9 00  ed
2ee0 : 20 e6 24 91 fe 20 f1 24  b0
2ee8 : c8 ad 15 2f d0 0c 18 98  b1
2ef0 : 69 06 8d 1d 68 a9 0b 8d  7e
2ef8 : 1f 68 a2 00 bd 1c 68 20  93
2f00 : e6 24 91 fe 20 f1 24 e8  30
2f08 : c8 d0 f1 60 43 42 4d 00  3c
2f10 : 00 00 00 00 00 00 8d 82  4c
2f18 : 33 8e 7e 33 a9 00 8d 81  6c
2f20 : 33 a9 00 a2 fd 8d 36 16  cd
```

```
2f28 : 8e 37 16 a9 00 a2 ea 85  d8
2f30 : fe 86 ff ad 7e 33 30 03  6f
2f38 : 4c 86 32 4c 55 33 a9 00  73
2f40 : a0 0f ae 7c 33 20 ba ff  c2
2f48 : a9 00 20 bd ff 20 c0 ff  b5
2f50 : 18 ad 7c 33 20 b1 ff a5  9f
2f58 : 90 0a 90 08 20 45 24 68  a0
2f60 : 68 4c c7 31 ad 7c 33 20  d2
2f68 : ae ff a9 0d a8 ae 7c 33  7a
2f70 : 20 ba ff a9 01 a2 7a a0  73
2f78 : 33 20 bd ff 20 c0 ff b0  94
2f80 : 09 60 a2 ff 8e 81 33 4c  bc
2f88 : 96 2f 20 cc ff a9 0d 38  49
2f90 : 20 c3 ff 4c af 2f 20 cc  aa
2f98 : ff a9 00 38 20 c3 ff ad  ee
2fa0 : 81 33 f0 0a c9 30 f0 06  26
2fa8 : 20 45 24 20 a7 31 60 a2  43
2fb0 : 00 20 c6 ff b0 e0 20 cf  a4
2fb8 : ff 8d 81 33 c9 00 f0 d6  53
2fc0 : c9 30 f0 d2 48 20 52 24  4f
2fc8 : 68 4c cf 2f 20 cf ff c9  44
2fd0 : 0d f0 09 20 d2 ff a5 90  81
2fd8 : 29 bf f0 f0 20 7a 13 4c  f6
2fe0 : 96 2f 8d 6e 33 ad 7f 33  44
2fe8 : 20 fb f9 8e 75 33 8d 76  6a
2ff0 : 33 ad 80 33 20 fb f9 8e  67
2ff8 : 78 33 8d 79 33 ad 7d 33  99
3000 : 20 fb f9 8d 73 33 a9 6d  a0
3008 : a2 33 85 89 86 8a a9 0d  54
3010 : 8d 7b 33 4c 16 30 a2 00  1f
3018 : 20 c9 ff 90 03 4c 8a 2f  4a
3020 : a0 00 b1 89 20 d2 ff c8  88
3028 : cc 7b 33 d0 f5 4c cc ff  8e
3030 : a9 31 20 e2 2f a5 90 29  8b
3038 : bf f0 08 a9 ff 8d 81 33  7f
3040 : 4c 8a 2f a2 0d 20 c6 ff  de
3048 : a0 00 20 cf ff 20 e6 24  cf
3050 : 91 fe 20 f1 24 c8 d0 f2  58
3058 : 4c 8a 2f 20 e6 24 b1 fe  0e
3060 : 20 f1 24 20 d2 ff c8 d0  78
3068 : f2 a0 00 20 e6 24 b1 fe  03
3070 : 20 f1 24 20 d2 ff 90 03  0b
3078 : 4c 8a 2f 20 cc ff a9 32  b1
3080 : 20 e2 2f a5 90 29 bf f0  c5
3088 : 03 20 82 2f 4c 8a 2f ae  55
3090 : 80 33 e8 8e 80 33 20 ee  b6
3098 : 24 a9 00 20 3e 2f 4c 30  84
30a0 : 30 a0 00 c6 ff 20 e6 24  de
30a8 : b1 fe 20 f1 24 d0 03 4c  8c
30b0 : 3c 31 c8 8d 7f 33 20 e6  48
30b8 : 24 b1 fe 20 f1 24 8d 80  f0
30c0 : 33 e6 ff 20 ee 24 a9 00  21
30c8 : 20 3e 2f 4c 30 30 20 ee  40
30d0 : 24 a9 00 20 3e 2f a2 0d  cf
30d8 : 20 c9 ff 90 03 4c 8a 2f  0a
30e0 : 20 5b 30 ad 81 33 30 0d  fc
30e8 : e6 ff ee 80 33 ad 80 33  a2
30f0 : cd 83 33 30 d9 4c 8a 2f  da
30f8 : 20 ee 24 a9 00 20 3e 2f  26
3100 : a2 0d 20 c9 ff 90 03 4c  93
3108 : 8a 2f a0 01 20 5b 30 ad  6b
3110 : 81 33 30 0c a0 00 20 e6  11
3118 : 24 b1 fe 20 f1 24 d0 03  62
3120 : 4c 3c 31 c8 20 e6 24 b1  1d
3128 : fe 20 f1 24 8d 80 33 e6  af
3130 : ff 4c f8 30 20 83 24 a9  9b
3138 : ff 8d 81 33 60 ad 82 33  a8
3140 : f0 fa 20 45 24 20 7d ff  97
3148 : 4c 4f 4f 50 2d 44 49 52  d8
3150 : 45 43 54 4f 52 59 20 21  e9
3158 : 0d 00 4c 34 31 88 20 e6  a5
3160 : 24 b1 fe 20 f1 24 f0 d4  ce
3168 : 20 45 24 20 7d ff 55 4e  02
3170 : 41 43 43 45 50 54 41 42  fe
3178 : 45 4c 20 53 45 43 54 4f  b4
3180 : 52 20 21 0d 00 4c 34 31  62
3188 : 20 45 24 20 7d ff 55 4e  22
3190 : 41 43 43 45 50 54 41 42  1e
3198 : 4c 45 20 54 52 41 43 4b  ec
31a0 : 20 21 0d 00 4c 34 31 20  00
31a8 : 7d ff 20 20 20 20 20 20  f5
31b0 : 20 20 20 20 20 20 44 49  93
31b8 : 53 4b 2d 45 52 52 4f 52  3e
31c0 : 20 21 0d 00 4c 34 31 48  70
31c8 : 20 7d ff 20 20 44 45 56  91
31d0 : 49 43 45 20 23 00 68 20  24
31d8 : fb f9 48 8a f0 03 20 d2  80
31e0 : ff 68 20 d2 ff 20 7d ff  6d
31e8 : 20 4e 4f 54 20 41 56 41  76
31f0 : 49 4c 41 42 4c 45 20 21  aa
31f8 : 0d 00 4c 34 31 20 45 24  10
3200 : 20 7d ff 49 27 4d 20 4e  02
3208 : 4f 54 20 41 42 4c 45 20  8d
3210 : 54 4f 20 48 41 4e 44 4c  4d
3218 : 45 20 43 50 2f 4d 2d 44  e3
3220 : 49 53 4b 53 0d 00 4c 34  bb
3228 : 31 20 45 24 20 5f 13 20  c9
3230 : 7d ff 20 20 20 20 20 41  bf
3238 : 54 54 45 4e 54 49 4f 4e  3b
3240 : 20 20 21 20 20 20 20 20  81
3248 : 41 54 54 45 4e 54 49 4f  bc
3250 : 4e 20 20 21 0d 20 54 52  a2
3258 : 41 43 4b 20 31 38 20 2f  c5
3260 : 20 53 45 43 54 4f 52 20  2d
3268 : 30 20 20 49 53 20 4d 41  c8
3270 : 4e 49 50 55 4c 41 54 45  cc
3278 : 44 20 21 00 20 68 13 20  e7
3280 : 36 24 20 b3 25 60 a9 12  67
3288 : a2 00 86 90 8e 81 33 8d  bb
3290 : 7f 33 8e 80 33 a0 00 20  d5
3298 : e6 24 91 fe c8 8a 91 fe  fa
32a0 : 20 f1 24 a9 00 20 3e 2f  4f
32a8 : 20 30 30 20 e6 24 a0 e0  c4
32b0 : b1 fe c9 43 d0 14 c8 b1  ef
32b8 : fe c9 42 d0 0d c8 b1 fe  21
32c0 : c9 4d d0 06 20 f1 24 4c  df
32c8 : fd 31 20 f1 24 ad 81 33  c0
32d0 : 10 01 60 38 a5 ff e9 ea  58
32d8 : cd 83 33 30 03 4c 3d 31  24
32e0 : ad 82 33 f0 51 a0 01 20  18
32e8 : e6 24 b1 fe 20 f1 24 cd  ea
32f0 : 83 33 30 17 ad 80 33 f0  89
32f8 : 03 4c 5d 31 20 29 32 a9  06
3300 : 01 20 e6 24 a0 01 91 fe  a6
3308 : 20 f1 24 88 20 e6 24 b1  68
3310 : fe 20 f1 24 c9 12 f0 1e  4c
3318 : c9 00 d0 03 4c 3c 31 ad  3d
3320 : 80 33 f0 03 4c 88 31 20  e4
3328 : 29 32 a9 12 a0 00 20 e6  6f
3330 : 24 91 fe 20 f1 24 e6 ff  bc
3338 : ad 82 33 f0 06 20 a1 30  5a
3340 : 4c cd 32 ae 80 33 e8 ec  f4
3348 : 83 33 d0 03 4c 3c 31 20  a5
3350 : 8f 30 4c cd 32 a9 00 a2  7a
3358 : ea 85 fe 86 ff a0 00 8c  b3
3360 : 80 33 ad 82 33 30 03 4c  8f
3368 : ce 30 4c f8 30 55 00 3a  a3
3370 : 31 33 20 30 20 00 00 20  8b
3378 : 00 00 23 00 08 00 00 12  e6
3380 : 00 00 01 13 a9 12 8d 7f  83
3388 : 33 a9 00 8d 81 33 8d 56  d6
3390 : 34 8d 58 34 a9 00 8d 57  a7
3398 : 34 ae 56 34 bd 44 34 8d  29
33a0 : 80 33 a9 00 20 3e 2f a2  1a
33a8 : 0d 20 c9 ff a9 00 20 d2  f8
33b0 : ff 90 03 4c 8a 2f ee 58  d0
33b8 : 34 ad 58 34 cd 28 15 b0  33
33c0 : 77 8d 38 16 20 39 16 a0  34
33c8 : 00 20 e6 24 b1 fc 20 f1  7e
33d0 : 24 20 d2 ff c8 c0 1e 90  e5
33d8 : f0 ee 57 34 ad 57 34 c9  96
33e0 : 08 b0 0d a9 00 20 d2 ff  05
33e8 : a9 00 20 d2 ff 4c b6 33  97
33f0 : ae 58 34 e8 ec 28 15 b0  ba
33f8 : 24 a9 12 20 d2 ff ee 56  0f
3400 : 34 ae 56 34 bd 44 34 20  b7
3408 : d2 ff 90 03 4c 8a 2f 20  74
3410 : cc ff a9 32 20 e2 2f 20  a3
3418 : 8a 2f 4c 94 33 a9 00 20  a0
3420 : d2 ff a9 ff 20 d2 ff 90  16
3428 : 03 4c 8a 2f 20 cc ff a9  96
3430 : 32 20 e2 2f 20 8a 2f 60  e5
3438 : a2 1e a9 00 20 d2 ff ca  82
3440 : d0 f8 f0 95 01 04 07 0a  dc
3448 : 0d 10 02 05 08 0b 0e 11  b2
3450 : 03 06 09 0c 0f 12 00 00  9c
3458 : 00 20 24 ca 20 b3 25 20  3f
3460 : 74 13 20 65 13 a0 4f ae  e3
3468 : d3 34 bd 81 34 20 d2 ff  84
3470 : e8 e0 50 90 02 a2 00 88  35
3478 : d0 f0 20 77 13 20 36 24  0b
3480 : 60 20 20 20 20 20 20 20  c0
3488 : 20 20 20 20 50 20 4f 20  48
3490 : 57 20 45 20 52 20 20 2d  4e
3498 : 20 20 31 20 32 20 38 20  5e
34a0 : 20 2f 20 20 56 31 2e 31  6e
34a8 : 20 20 20 20 20 20 20 20  a8
34b0 : 2e 2e 2e 2e 2e 20 50 4c  04
34b8 : 45 41 53 45 20 54 41 4b  5c
34c0 : 45 20 59 4f 55 52 20 43  45
34c8 : 48 4f 49 43 45 20 2e 2e  dd
34d0 : 2e 2e 2e 00 a9 02 a2 54  7f
34d8 : a0 01 20 a3 13 20 5f 13  4b
34e0 : a2 00 8e d3 34 20 59 34  b2
34e8 : 20 36 24 20 5f 13 a2 01  4c
34f0 : a0 0a 20 c3 25 20 7d ff  5f
34f8 : 12 05 20 46 31 20 20 20  33
3500 : 20 20 20 20 20 92 20 20  94
3508 : 20 20 20 45 58 54 45 4e  c3
3510 : 44 45 44 20 44 49 52 45  6e
3518 : 43 54 4f 52 59 00 a2 03  ca
3520 : 20 c3 25 20 7d ff 12 1f  ce
3528 : 20 46 32 20 20 20 20 20  c0
3530 : 20 20 20 92 20 20 20 20  7f
3538 : 20 53 41 56 45 20 41 4e  14
3540 : 44 20 52 45 53 54 4f 52  8b
3548 : 45 20 44 49 52 45 43 54  dd
3550 : 4f 52 59 2d 54 52 41 43  28
3558 : 4b 53 00 a2 05 20 c3 25  4c
3560 : 20 7d ff 12 95 20 46 33  5b
3568 : 20 20 20 20 20 20 20 20  68
3570 : 92 20 20 20 20 20 4d 4f  f5
3578 : 44 49 46 59 20 42 41 4d  d1
3580 : 20 41 4e 44 20 44 49 52  4b
3588 : 45 43 54 4f 52 59 2d 45  9d
3590 : 4e 54 52 49 45 53 00 a2  fa
3598 : 07 20 c3 25 20 7d ff 12  57
35a0 : 97 20 46 34 20 20 20 20  23
35a8 : 20 20 20 20 92 20 20 20  cf
35b0 : 20 20 50 52 4f 54 45 43  72
35b8 : 54 20 41 4e 44 20 52 45  50
35c0 : 4c 45 41 53 45 20 44 49  63
35c8 : 53 4b 53 20 41 4e 44 20  72
35d0 : 46 49 4c 45 53 00 a2 09  48
35d8 : 20 c3 25 20 7d ff 12 9f  87
35e0 : 20 46 35 20 20 20 20 20  39
35e8 : 20 20 20 92 20 20 20 20  37
35f0 : 20 53 4f 52 54 20 26 20  f7
```

Listing 1. »Power 128/v.1.1« (Fortsetzung)

```
35f8 : 4d 4f 44 49 46 59 20 44   5f
3600 : 49 52 45 43 54 4f 52 59   e8
3608 : 00 a2 0b 20 c3 25 20 7d   01
3610 : ff 12 9a 20 46 36 20 20   9a
3618 : 20 20 20 20 20 20 92 20   e2
3620 : 20 20 20 20 50 52 49 4e   b6
3628 : 54 20 53 45 4c 45 43 54   ae
3630 : 45 44 20 44 49 52 45 43   eb
3638 : 54 4f 52 59 00 a2 0d 20   7d
3640 : c3 25 20 7d ff 12 9b 20   8d
3648 : 46 37 20 20 20 20 20 20   fa
3650 : 20 20 92 20 20 20 20 20   ed
3658 : 50 52 49 4e 54 20 44 49   d7
3660 : 53 43 2d 4c 41 42 45 4c   fe
3668 : 53 00 a2 0f 20 c3 25 20   3b
3670 : 7d ff 12 81 20 46 38 20   f7
3678 : 20 20 20 20 20 20 20 92   5d
3680 : 20 20 20 20 20 42 4f 4f   ac
3688 : 54 20 2d 20 48 45 4c 50   bc
3690 : 00 a2 11 20 c3 25 20 7d   0b
3698 : ff 12 9e 20 48 45 4c 50   cc
36a0 : 20 20 20 20 20 20 92 20   6a
36a8 : 20 20 20 20 43 42 4d 20   a0
36b0 : 2d 20 53 54 41 4e 44 41   67
36b8 : 52 44 2d 44 49 52 45 43   c3
36c0 : 54 4f 52 59 00 a2 15 20   25
36c8 : c3 25 20 7d ff 12 96 20   01
36d0 : 45 53 43 20 20 20 20 20   58
36d8 : 20 20 00 20 68 13 20 7d   a7
36e0 : ff 92 20 20 20 20 20 45   43
36e8 : 58 49 54 20 50 52 4f 47   61
36f0 : 52 41 4d 4d 20 41 4e 44   ae
36f8 : 20 52 45 53 54 41 52 54   3e
3700 : 20 53 59 53 54 45 4d 00   2f
3708 : 20 a9 14 a9 00 8d 20 27   72
3710 : ad 90 13 cd a1 13 f0 03   00
3718 : 20 e8 14 20 e4 ff c9 81   2e
3720 : f0 03 4c 2b 37 4c ed 37   06
3728 : 4c cb 37 c9 82 f0 03 4c   b5
3730 : 38 37 4c ca 3d 4c cb 37   44
3738 : c9 83 f0 03 4c 45 37 4c   c4
3740 : c8 45 4c cb 37 c9 84 f0   ed
3748 : 03 4c 52 37 4c fa 41 4c   27
3750 : cb 37 c9 85 f0 03 4c 5f   f1
3758 : 37 4c 41 44 4c cb 37 c9   22
3760 : 86 f0 03 4c 6c 37 4c b9   ce
3768 : 44 4c cb 37 c9 87 f0 03   4f
3770 : 4c 79 37 4c 32 45 4c cb   e6
3778 : 37 c9 88 f0 03 4c 86 37   ef
3780 : 4c cf 4b 4c cb 37 c9 89   c1
3788 : f0 03 4c 93 37 ea ea ea   cc
3790 : 4c cb 37 c9 8a f0 03 4c   9e
3798 : a0 37 4c 1d 41 4c cb 37   9f
37a0 : c9 1b f0 03 4c ad 37 4c   3b
37a8 : d5 37 4c cb 37 ee d3 34   48
37b0 : ae d3 34 e0 50 90 05 a2   54
37b8 : 00 8e d3 34 20 59 34 a2   5e
37c0 : a0 a0 ff 88 d0 fd ca d0   8b
37c8 : f8 a9 00 c9 1b d0 03 4c   ab
37d0 : d5 37 4c 1b 37 20 24 ca   52
37d8 : 20 5f 13 20 f1 24 a0 4e   d0
37e0 : b9 a8 ce 99 00 10 88 10   97
37e8 : f7 20 f1 24 60 a9 93 a2   d7
37f0 : 55 a0 02 20 a3 13 20 03   73
37f8 : 13 a2 0b a9 20 20 d2 ff   a3
3800 : ca d0 f8 20 7d ff 43 4f   f8
3808 : 4d 50 4c 45 54 45 20 44   b2
3810 : 49 52 45 43 54 4f 52 59   f8
3818 : 20 49 4e 46 4f 52 4d 41   78
3820 : 54 49 4f 4e 00 a2 12 a9   67
3828 : 20 20 d2 ff ca d0 f8 20   64
3830 : 30 13 20 e4 ff c9 81 f0   c5
3838 : 03 4c 5f 3c 20 36 24 20   45
3840 : 5f 13 ad 91 13 f0 38 a2   a5
3848 : 05 a0 0a 20 c3 25 20 b3   71
3850 : 29 a9 07 a2 35 20 8c 29   3d
3858 : 20 36 24 a2 0a a0 0a 20   ff
3860 : c3 25 a9 3c 20 e8 29 a9   e9
3868 : 4b a2 0d a0 00 20 61 29   35
3870 : a9 0f a2 0f a0 3c 20 a2   dd
3878 : 29 20 36 24 20 5f 13 20   4d
3880 : c2 24 20 44 17 ad 81 33   30
3888 : 10 03 4c 57 3c 20 2c 15   b7
3890 : 20 5f 13 a9 d4 a2 56 a0   57
3898 : 02 20 a3 13 20 a9 14 a2   da
38a0 : 01 20 c9 ff a2 05 a9 20   5d
38a8 : 20 d2 ff ca d0 f8 20 39   52
38b0 : 18 a2 00 bd 97 3c 20 d2   53
38b8 : ff e8 e0 09 d0 f5 20 44   4a
38c0 : 18 20 c8 25 a2 05 a9 20   f8
38c8 : 20 d2 ff ca d0 f8 20 39   72
38d0 : 18 a2 09 bd 97 3c 20 d2   b5
38d8 : ff e8 e0 0e d0 f5 20 44   0b
38e0 : 18 20 d8 25 a2 06 a9 20   24
38e8 : 20 d2 ff ca d0 f8 20 39   92
38f0 : 18 a2 0e bd 97 3c 20 d2   16
38f8 : ff e8 e0 1d d0 f5 20 44   0d
3900 : 18 20 02 26 20 7a 13 a2   d5
3908 : 05 a9 20 20 d2 ff ca d0   e8
3910 : f8 20 39 18 a2 1d bd 97   a3
3918 : 3c 20 d2 ff e8 e0 2c d0   01
3920 : f5 20 56 26 a2 01 20 c9   c6
3928 : ff 20 44 18 20 c8 26 20   6d
3930 : 58 17 a2 1b a9 20 20 d2   e2
3938 : ff ca d0 f8 20 39 18 a2   61
3940 : 2d bd 97 3c 20 d2 ff e8   24
3948 : e0 3d d0 f5 20 44 18 20   7e
3950 : a5 17 20 7a 13 a2 05 a9   86
3958 : 20 20 d2 ff ca d0 f8 20   94
3960 : 39 18 a2 3d bd 97 3c 20   bf
3968 : d2 ff e8 e0 4d d0 f5 20   03
3970 : 44 18 20 e2 26 ad 91 13   61
3978 : f0 24 a2 04 a9 20 20 d2   65
3980 : ff ca d0 f8 a2 00 bd b8   ca
3988 : 3d 20 d2 ff e8 e0 12 d0   09
3990 : f5 a2 05 a9 20 20 d2 ff   9b
3998 : ca d0 f8 4c a8 39 a2 1b   a7
39a0 : a9 20 20 d2 ff ca d0 f8   47
39a8 : 20 39 18 a2 4d bd 97 3c   59
39b0 : 20 d2 ff e8 e0 5d d0 f5   7e
39b8 : 20 44 18 20 f0 17 20 7a   42
39c0 : 13 a2 05 a9 20 20 d2 ff   e9
39c8 : ca d0 f8 20 39 18 a2 5d   d6
39d0 : bd 97 3c 20 d2 ff e8 e0   fe
39d8 : 6c d0 f5 20 44 18 ae 28   3e
39e0 : 15 ca a0 00 8a a2 00 20   80
39e8 : 24 25 ad 91 13 f0 20 a2   bb
39f0 : 05 a9 20 20 d2 ff ca d0   d0
39f8 : f8 a9 c3 a0 67 a2 0e 20   ce
3a00 : 88 28 a2 08 a9 20 20 d2   08
3a08 : ff ca d0 f8 4c 19 3a a2   7b
3a10 : 1b a9 20 20 d2 ff ca d0   06
3a18 : f8 20 39 18 a2 6d bd 97   2d
3a20 : 3c 20 d2 ff e8 e0 7d d0   4e
3a28 : f5 20 44 18 20 0c 18 ad   60
3a30 : 91 13 f0 0f 20 7a 13 20   cb
3a38 : 7a 13 a2 4b a9 d1 a0 67   c8
3a40 : 20 88 28 20 7a 13 20 7a   68
3a48 : 13 a2 05 a9 20 20 d2 ff   71
3a50 : ca d0 f8 20 39 18 a2 48   34
3a58 : a9 2d 20 d2 ff ca d0 f8   86
3a60 : a2 7d bd 97 3c 20 d2 ff   33
3a68 : e8 e0 d2 d0 f5 a2 48 a9   78
3a70 : 2d 20 d2 ff ca d0 f8 20   b9
3a78 : 7a 13 a2 00 bd 69 3d 20   81
3a80 : d2 ff e8 e0 4f d0 f5 ad   57
3a88 : 91 13 d0 06 20 44 18 20   5d
3a90 : 14 ca a9 00 8d 38 16 8d   82
3a98 : 96 3c 20 e4 ff d0 09 a9   ef
3aa0 : 00 0d 96 3c d0 f4 f0 1d   07
3aa8 : c9 1b d0 03 4c 23 3c c9   f6
3ab0 : 20 d0 ec 2d 96 3c d0 08   b8
3ab8 : a9 20 8d 96 3c 4c 9a 3a   ac
3ac0 : a9 00 8d 96 3c ee 38 16   e8
3ac8 : 20 39 16 a5 fc d0 07 a5   7d
3ad0 : fd d0 03 4c 23 3c 20 39   87
3ad8 : 18 a2 00 bd 69 3d 20 d2   a0
3ae0 : ff e8 e0 06 d0 f5 20 44   12
3ae8 : 18 a9 10 a0 02 20 6c 16   ec
3af0 : 20 39 18 a2 02 bd 69 3d   35
3af8 : 20 d2 ff e8 e0 06 d0 f5   0c
3b00 : 20 44 18 20 ff 26 20 39   71
3b08 : 18 a2 03 bd 69 3d 20 d2   90
3b10 : ff e8 e0 06 d0 f5 20 44   42
3b18 : 18 20 34 27 20 39 18 a2   a4
3b20 : 03 bd 69 3d 20 d2 ff e8   6e
3b28 : e0 06 d0 f5 20 44 18 20   c3
3b30 : 70 27 20 39 18 a2 03 bd   81
3b38 : 69 3d 20 d2 ff e8 e0 06   79
3b40 : d0 f5 20 44 18 ad 74 25   a7
3b48 : 48 a9 03 8d 74 25 a0 01   cc
3b50 : 20 04 25 8d 80 33 8e 7f   48
3b58 : 33 48 8a a2 00 a0 00 20   ec
3b60 : 24 25 20 39 18 a2 2e bd   11
3b68 : 69 3d 20 d2 ff e8 e0 30   fd
3b70 : d0 f5 20 44 18 a2 00 a0   a3
3b78 : 00 68 20 24 25 68 8d 74   ee
3b80 : 25 20 39 18 a2 04 bd 69   1b
3b88 : 3d 20 d2 ff e8 e0 07 d0   dd
3b90 : f5 20 44 18 ad 1f 27 c9   ad
3b98 : 02 d0 29 ad 74 25 48 a9   e7
3ba0 : 06 8d 74 25 20 89 27 20   5a
3ba8 : 39 18 a2 2e bd 69 3d 20   b8
3bb0 : d2 ff e8 e0 30 d0 f5 20   7a
3bb8 : 44 18 20 c8 27 68 8d 74   fe
3bc0 : 25 4c d4 3b 20 39 18 a2   19
3bc8 : 35 bd 69 3d 20 d2 ff e8   48
3bd0 : e0 41 d0 f5 20 39 18 a2   b5
3bd8 : 41 bd 69 3d 20 d2 ff e8   64
3be0 : e0 44 d0 f5 20 44 18 ad   b5
3be8 : 1f 27 c9 04 d0 0f a0 14   be
3bf0 : 20 04 25 a2 00 a0 00 20   f5
3bf8 : 24 25 4c 0d 3c 20 39 18   3d
3c00 : a2 44 bd 69 3d 20 d2 ff   81
3c08 : e8 e0 48 d0 f5 20 39 18   02
3c10 : a2 48 bd 69 3d 20 d2 ff   93
3c18 : e8 e0 4f d0 f5 20 44 18   00
3c20 : 4c 9a 3a a2 05 a9 20 20   fb
3c28 : d2 ff ca d0 f8 20 39 18   6c
3c30 : a2 48 a9 2d 20 d2 ff ca   35
3c38 : d0 f8 20 44 18 20 7a 13   a8
3c40 : ad 91 13 f0 06 ad 9d 13   03
3c48 : 20 d2 ff 20 de 14 a9 93   32
3c50 : a2 55 a0 02 20 a3 13 a9   c4
3c58 : 00 8d 96 3c 4c 89 3c c9   e2
3c60 : 87 f0 03 4c 6e 3c 20 ad   4e
3c68 : 13 a9 00 4c 89 3c c9 88   8c
3c70 : f0 03 4c 7d 3c 20 2f 14   4e
3c78 : a9 00 4c 89 3c c9 8a f0   84
3c80 : 03 4c 89 3c 20 78 25 a9   41
3c88 : 00 c9 1b d0 03 4c 93 3c   a7
3c90 : 4c 32 38 4c d4 34 00 48   0c
3c98 : 45 41 44 45 52 20 3a 20   87
3ca0 : 49 44 20 3a 20 44 49 53   4b
3ca8 : 4b 2d 46 4f 52 4d 41 54   43
3cb0 : 20 20 3a 20 42 4f 4f 54   f7
3cb8 : 45 44 20 42 4c 4f 43 4b   53
3cc0 : 53 20 3a 20 42 4c 4f 43   00
3cc8 : 4b 53 20 46 52 45 45 20   32
3cd0 : 20 3a 20 20 44 49 53 4b   8c
3cd8 : 20 53 54 41 54 55 53 20   5d
3ce0 : 20 20 3a 20 42 4c 4f 43   ed
```

```
3ce8 : 4b 53 20 55 53 45 44 20  40
3cf0 : 20 3a 20 20 46 49 4c 45  a4
3cf8 : 53 20 54 4f 54 41 4c 20  1b
3d00 : 20 20 3a 20 42 4c 4f 43  0d
3d08 : 4b 53 20 54 4f 54 41 4c  c5
3d10 : 20 3a 20 20 0d 20 20 20  ec
3d18 : 20 21 20 46 49 4c 45 2d  00
3d20 : 4e 41 4d 45 20 20 20 20  cf
3d28 : 20 20 20 20 20 21 20 54  99
3d30 : 59 50 20 21 20 53 54 20  0c
3d38 : 21 20 42 4c 4b 53 20 21  96
3d40 : 20 54 20 3a 20 53 20 21  39
3d48 : 20 53 54 41 52 54 2d 41  4e
3d50 : 44 52 2d 45 4e 44 20 21  7b
3d58 : 20 52 45 4c 2d 4c 45 4e  63
3d60 : 20 20 21 0d 20 20 20 20  3e
3d68 : 20 20 20 20 20 21 20 20  70
3d70 : 20 20 20 20 20 20 20 20  70
3d78 : 20 20 20 20 20 20 20 20  78
3d80 : 20 21 20 20 20 20 20 21  03
3d88 : 20 20 20 20 21 20 20 20  98
3d90 : 20 20 20 21 20 20 20 3a  e5
3d98 : 20 20 20 21 20 20 20 20  b8
3da0 : 20 20 20 3a 20 20 20 20  e4
3da8 : 20 20 20 21 20 20 20 20  c8
3db0 : 20 20 20 20 20 20 21 0d  8e
3db8 : 44 41 54 45 20 4f 46 20  30
3dc0 : 43 52 45 41 54 49 4f 4e  0f
3dc8 : 20 3a a9 25 a2 57 a0 02  80
3dd0 : 20 a3 13 20 03 13 a2 09  f0
3dd8 : a9 20 20 d2 ff ca d0 f8  7f
3de0 : 20 7d ff 53 41 56 45 20  45
3de8 : 41 4e 44 20 52 45 53 54  ab
3df0 : 4f 52 45 20 44 49 52 45  20
3df8 : 43 54 4f 52 59 20 2d 20  0f
3e00 : 54 52 41 43 4b 53 00 a2  cb
3e08 : 0f a9 20 20 d2 ff ca d0  f2
3e10 : f8 20 30 13 20 e4 ff c9  44
3e18 : 81 f0 03 4c ad 3e 20 b9  1c
3e20 : 25 20 6e 13 20 7d ff 49  d4
3e28 : 4e 53 45 52 54 20 44 49  a5
3e30 : 53 4b 20 54 4f 20 42 45  45
3e38 : 20 53 41 56 45 44 20 41  96
3e40 : 4e 44 20 48 49 54 20 20  b9
3e48 : 3c 20 46 20 31 20 3e 00  37
3e50 : 20 e4 ff c9 1b f0 04 c9  f8
3e58 : 81 d0 f5 c9 1b d0 03 4c  d5
3e60 : f8 40 20 c2 24 20 3e 17  43
3e68 : 20 5f 13 ad 81 33 30 35  8f
3e70 : 20 b9 25 20 71 13 20 7d  e5
3e78 : ff 43 4f 4e 54 45 4e 53  06
3e80 : 20 4f 46 20 44 49 52 45  40
3e88 : 43 54 4f 52 59 2d 54 52  08
3e90 : 41 43 4b 20 53 54 4f 52  03
3e98 : 45 44 20 49 4e 20 4d 45  d6
3ea0 : 4d 4f 52 59 00 20 68 13  1d
3ea8 : a9 00 4c ee 40 c9 83 f0  84
3eb0 : 03 4c 3b 3f 20 b9 25 20  35
3eb8 : 6e 13 20 7d ff 49 4e 53  92
3ec0 : 45 52 54 20 44 49 53 4b  ba
3ec8 : 20 57 49 54 48 20 53 41  c6
3ed0 : 56 45 2d 46 49 4c 45 20  29
3ed8 : 41 4e 44 20 48 49 54 20  b6
3ee0 : 20 3c 20 46 20 33 20 3e  88
3ee8 : 00 20 68 13 20 e4 ff c9  32
3ef0 : 1b f0 04 c9 83 d0 f5 c9  e8
3ef8 : 1b d0 03 4c f8 40 20 79  cb
3f00 : 2a a9 00 a2 00 a0 ea 20  44
3f08 : d5 ff 90 03 4c d8 40 20  2e
3f10 : b9 25 20 fb 40 20 71 13  d4
3f18 : 20 7d ff 20 20 49 53 20  d5
3f20 : 4c 4f 41 44 45 44 20 54  8c
3f28 : 4f 20 4d 45 4d 4f 52 59  cf
3f30 : 00 20 68 13 a9 00 85 90  8f
3f38 : 4c ee 40 c9 84 f0 03 4c  b9
3f40 : c4 3f 20 b9 25 20 6e 13  16
3f48 : 20 7d ff 49 4e 53 45 52  89
3f50 : 54 20 44 49 53 4b 20 54  a7
3f58 : 4f 20 42 45 20 57 52 49  89
3f60 : 54 54 45 4e 20 41 4e 44  c7
3f68 : 20 48 49 54 20 20 3c 20  bd
3f70 : 46 20 34 20 3e 00 20 68  0d
3f78 : 13 20 e4 ff c9 1b f0 04  16
3f80 : c9 84 d0 f5 c9 1b d0 03  3d
3f88 : 4c f8 40 20 c2 24 a9 12  7d
3f90 : 8d 7f 33 20 4b 17 20 5f  5a
3f98 : 13 ad 81 33 30 1e 20 71  a0
3fa0 : 13 20 7d ff 44 49 52 45  85
3fa8 : 43 54 4f 52 59 2d 54 52  28
3fb0 : 41 43 4b 20 57 52 49 54  40
3fb8 : 54 45 4e 00 20 68 13 a9  27
3fc0 : 00 4c ee 40 c9 87 f0 03  4d
3fc8 : 4c d3 3f 20 ad 13 a9 00  ec
3fd0 : 4c ee 40 c9 86 f0 03 4c  71
3fd8 : e2 40 20 b9 25 20 6e 13  4d
3fe0 : 20 7d ff 49 4e 53 45 52  21
3fe8 : 54 20 44 49 53 4b 2c 20  07
3ff0 : 4f 4e 20 57 48 49 43 48  c6
3ff8 : 20 44 49 52 45 43 54 4f  35
4000 : 52 59 2d 54 52 41 43 4b  a8
4008 : 20 28 53 54 4f 52 45 44  c1
4010 : 20 49 4e 20 4d 45 4d 4f  3f
4018 : 52 59 29 0d 20 20 20 20  c7
4020 : 20 20 20 20 20 20 53 48  3d
4028 : 41 4c 4c 20 42 45 20 53  1c
4030 : 41 56 45 44 20 20 20 2d  54
4038 : 20 20 54 48 45 4e 20 48  5e
4040 : 49 54 20 20 3c 20 46 20  de
4048 : 36 20 3e 00 20 68 13 20  f0
4050 : e4 ff c9 1b f0 04 c9 86  6d
4058 : d0 f5 c9 1b d0 03 4c f8  41
4060 : 40 20 79 2a a2 00 a0 ea  d6
4068 : 86 fe 84 ff 18 a9 00 69  30
4070 : 01 aa a9 fd 69 00 a8 a9  7d
4078 : fe 20 d8 ff a5 90 f0 1e  9b
4080 : 20 5f 13 20 45 24 20 7d  0a
4088 : ff 46 49 4c 45 20 2d 20  d1
4090 : 45 52 52 4f 52 20 21 0d  42
4098 : 00 20 83 24 f0 3a 20 b9  e3
40a0 : 25 20 71 13 20 7d ff 44  0b
40a8 : 49 52 45 43 54 4f 52 59  90
40b0 : 2d 54 52 41 43 4b 20 49  66
40b8 : 4e 20 4d 45 4d 4f 52 59  5e
40c0 : 20 49 53 20 53 41 56 45  81
40c8 : 44 20 4f 4e 20 46 49 4c  ac
40d0 : 45 20 20 20 00 20 fb 40  a3
40d8 : 20 68 13 a9 00 85 90 4c  2d
40e0 : ee 40 c9 8a f0 03 4c ee  e8
40e8 : 40 20 78 25 a9 00 c9 1b  f3
40f0 : d0 03 4c f8 40 4c 14 3e  a7
40f8 : 4c d4 34 20 6e 13 a2 00  ca
4100 : bd 0d 41 20 d2 ff e8 ec  43
4108 : a9 2a d0 f4 60 53 41 56  eb
4110 : 45 44 20 54 52 41 43 4b  dd
4118 : 20 20 23 20 31 a9 66 a2  54
4120 : 58 a0 02 20 a3 13 20 03  a6
4128 : 13 a2 08 a9 20 20 d2 ff  12
4130 : ca d0 f8 20 7d ff 20 20  3d
4138 : 20 20 43 20 42 20 4d 20  d8
4140 : 20 20 2d 20 20 53 54 41  30
4148 : 4e 44 41 52 44 20 2d 20  8d
4150 : 44 49 52 45 43 54 4f 52  2f
4158 : 59 20 20 20 20 00 a2 0c  72
4160 : a9 20 20 d2 ff ca d0 f8  07
4168 : 20 30 13 20 e4 ff c9 81  e2
4170 : f0 03 4c 7a 41 a9 50 4c  7f
4178 : c5 41 c9 87 f0 03 4c 89  ac
4180 : 41 20 ad 13 a9 00 4c c5  f6
4188 : 41 c9 82 f0 03 4c 95 41  d8
4190 : a9 55 4c c5 41 c9 88 f0  16
4198 : 03 4c a4 41 20 2f 14 a9  32
41a0 : 00 4c c5 41 c9 83 f0 03  e2
41a8 : 4c b0 41 a9 53 4c c5 41  03
41b0 : c9 84 f0 03 4c bc 41 a9  5b
41b8 : 52 4c c5 41 c9 8a f0 03  85
41c0 : 4c c5 41 a9 20 c9 1b d0  d3
41c8 : 03 4c f7 41 c9 00 f0 22  bc
41d0 : 48 a9 2a 8d b7 12 a9 3d  56
41d8 : 8d b8 12 68 8d b9 12 a9  95
41e0 : 03 8d 10 01 8d 11 01 a2  79
41e8 : 01 20 c9 ff 20 75 25 20  ee
41f0 : cc ff a9 00 4c 6b 41 4c  e4
41f8 : d4 34 a9 a7 a2 59 a0 02  c1
4200 : 20 a3 13 ad 90 13 cd a1  88
4208 : 13 f0 03 20 e8 14 20 03  0e
4210 : 13 a2 0b a9 20 20 d2 ff  bb
4218 : ca d0 f8 20 7d ff 50 52  4a
4220 : 4f 54 45 43 54 20 26 20  72
4228 : 52 45 4c 45 41 53 45 20  dd
4230 : 20 20 20 20 44 49 53 4b  df
4238 : 53 20 26 20 46 49 4c 45  93
4240 : 53 00 a2 0d a9 20 20 d2  9f
4248 : ff ca d0 f8 20 30 13 20  10
4250 : e4 ff c9 81 f0 03 4c 8b  46
4258 : 42 20 ed 43 f0 25 a9 42  f1
4260 : 20 d2 ff 20 cc ff a9 32  c5
4268 : 20 e2 2f 20 8a 2f ad 81  a5
4270 : 33 30 10 20 2e 44 20 7d  44
4278 : ff 50 52 4f 54 45 43 54  43
4280 : 45 44 00 20 68 13 a9 00  b1
4288 : 4c 13 43 c9 82 f0 03 4c  bc
4290 : d8 42 20 ed 43 f0 39 a9  43
4298 : 41 20 d2 ff 20 cc ff a9  5a
42a0 : 27 a2 44 a0 07 85 89 86  0d
42a8 : 8a 8c 7b 33 20 16 30 a9  84
42b0 : 32 20 e2 2f a9 32 20 e2  03
42b8 : 2f 20 8a 2f ad 81 33 30  94
42c0 : 0f 20 2e 44 20 7d ff 52  86
42c8 : 45 4c 45 41 53 45 44 00  1d
42d0 : 20 68 13 a9 00 4c 13 43  54
42d8 : c9 87 f0 03 4c e7 42 20  4f
42e0 : ad 13 a9 00 4c 13 43 c9  7f
42e8 : 8a f0 03 4c f6 42 20 78  28
42f0 : 25 a9 00 4c 13 43 c9 83  ed
42f8 : f0 03 4c 06 43 20 20 43  7a
4300 : 4c fa 41 4c 13 43 c9 84  1f
4308 : f0 03 4c 13 43 20 20 43  2c
4310 : 4c fa 41 c9 1b d0 03 4c  40
4318 : 1d 43 4c 4f 42 4c d4 34  16
4320 : a9 e8 a2 5a a0 02 20 a3  13
4328 : 13 20 ed 43 d0 01 60 20  06
4330 : 8a 2f 20 44 17 ad 81 33  2e
4338 : 10 03 4c e7 43 20 2c 15  ea
4340 : 20 5f 13 20 72 2b 20 7a  cf
4348 : 13 20 aa 2a a9 00 8d 38  9c
4350 : 16 ad 38 16 8d d8 2c a9  b1
4358 : 06 a2 00 20 2e ca 20 5f  2c
4360 : 13 a9 03 a2 10 a0 03 20  af
4368 : d7 2b 20 e4 ff c9 2b f0  56
4370 : 03 4c 8e 43 20 b3 25 ae  37
4378 : 38 16 e0 90 f0 05 ad b0  55
4380 : 2c f0 05 a9 00 8d 38 16  14
4388 : 4c 51 43 4c dd 43 c9 83  fd
4390 : f0 03 4c ae 43 48 20 9f  21
4398 : 18 a0 08 20 6f 29 20 1e  03
43a0 : 2d 20 36 24 68 aa 20 c7  db
43a8 : 1b f0 1e 4c dd 43 c9 84  75
43b0 : f0 03 4c d1 43 48 20 9f  a5
```

Listing 1. »Power 128/v.1.1« (Fortsetzung)

```
43b8 : 18 a0 08 20 6f 29 20 1e   23
43c0 : 2d 20 36 24 68 aa 20 c7   fb
43c8 : 1b 20 b1 2c a9 00 4c dd   6d
43d0 : 43 c9 55 f0 03 4c dd 43   fc
43d8 : 20 52 17 a9 1b c9 1b d0   2a
43e0 : 03 4c e7 43 4c 6a 43 20   d1
43e8 : 5f 13 a9 00 60 20 4f 18   b0
43f0 : c9 1b f0 f3 20 5f 13 20   8b
43f8 : c2 24 a9 30 8d 24 15 8d   a6
4400 : 25 15 a9 32 8d 26 15 a9   12
4408 : 12 a2 00 8d 7f 33 8e 80   ea
4410 : 33 20 3e 2f a9 31 20 e2   33
4418 : 2f 20 0c 15 20 0c 15 a2   f9
4420 : 0d 20 c9 ff a9 ff 60 4d   66
4428 : 2d 57 01 01 01 41 20 b9   6f
4430 : 25 20 71 13 20 7d ff 44   9b
4438 : 49 53 4b 20 49 53 20 00   b1
4440 : 60 a9 29 a2 5c a0 02 20   27
4448 : a3 13 20 03 13 a2 0e a9   af
4450 : 20 20 d2 ff ca d0 f8 20   8c
4458 : 7d ff 53 4f 52 54 20 26   28
4460 : 20 4d 4f 44 49 46 59 20   f0
4468 : 44 49 52 45 43 54 4f 52   47
4470 : 59 00 a2 16 a9 20 20 d2   f6
4478 : ff ca d0 f8 20 30 13 20   40
4480 : e4 ff c9 20 f0 03 4c 91   56
4488 : 44 20 67 1b a9 00 4c ac   3f
4490 : 44 c9 87 f0 03 4c a0 44   56
4498 : 20 ad 13 a9 00 4c ac 44   26
44a0 : c9 8a f0 03 4c ac 44 20   c6
44a8 : 78 25 a9 00 c9 1b d0 03   dc
44b0 : 4c b6 44 4c 7f 44 4c d4   e7
44b8 : 34 a9 ab a2 5e a0 02 20   33
44c0 : a3 13 20 03 13 a2 0e a9   27
44c8 : 20 20 d2 ff ca d0 f8 20   04
44d0 : 7d ff 50 52 49 4e 54 20   44
44d8 : 53 45 4c 45 43 54 45 44   fe
44e0 : 20 44 49 52 45 43 54 4f   1d
44e8 : 52 59 00 a2 15 a9 20 20   9b
44f0 : d2 ff ca d0 f8 20 30 13   06
44f8 : 20 e4 ff c9 20 f0 03 4c   f2
4500 : 0a 45 20 77 1e a9 00 4c   6c
4508 : 25 45 c9 87 f0 03 4c 19   be
4510 : 45 20 ad 13 a9 00 4c 25   49
4518 : 45 c9 8a f0 03 4c 25 45   b4
4520 : 20 78 25 a9 00 c9 1b d0   57
4528 : 03 4c 2f 45 4c f8 44 4c   fc
4530 : d4 34 a9 ab a2 5e a0 02   a2
4538 : 20 a3 13 20 03 13 a2 12   6a
4540 : a9 20 20 d2 ff ca d0 f8   e7
4548 : 20 7d ff 50 52 49 4e 54   82
4550 : 20 20 44 49 53 4b 20 2d   25
4558 : 20 4c 41 42 45 4c 53 00   3b
4560 : a2 15 a9 20 20 d2 ff ca   29
4568 : d0 f8 20 30 13 20 e4 ff   88
4570 : c9 20 f0 03 4c 8d 45 8d   47
4578 : 51 20 a2 00 8e 4b 20 8e   63
4580 : 9d 1e 20 77 1e a9 00 8d   6e
4588 : 51 20 4c a8 45 c9 87 f0   b4
4590 : 03 4c 9c 45 20 ad 13 a9   98
4598 : 00 4c a8 45 c9 8a f0 03   4c
45a0 : 4c a8 45 20 78 25 a9 00   ed
45a8 : a2 00 8e 4b 20 8e 9d 1e   81
45b0 : c9 1b d0 03 4c ba 45 4c   e4
45b8 : 6d 45 a2 00 8e 51 20 8e   82
45c0 : 4b 20 8e 9d 1e 4c d4 34   73
45c8 : a9 7e a2 61 a0 02 20 a3   67
45d0 : 13 20 03 13 a2 0c a9 20   88
45d8 : 20 d2 ff ca d0 f8 20 7d   0b
45e0 : ff 4d 4f 44 49 46 59 20   4f
45e8 : 20 42 41 4d 20 20 26 20   ff
45f0 : 20 44 49 52 45 43 54 4f   2d
45f8 : 52 59 20 2d 20 45 4e 54   b3
4600 : 52 49 45 53 00 a2 0c a9   4b
4608 : 20 20 d2 ff ca d0 f8 20   44
4610 : 30 13 20 e4 ff c9 81 f0   a5
4618 : 03 4c 3f 46 8d df 47 a9   22
4620 : bf a2 62 a0 02 20 a3 13   b3
4628 : 20 fe 47 20 5f 13 a9 7e   cf
4630 : a2 61 a0 02 20 a3 13 a9   aa
4638 : 00 8d df 47 4c a2 46 c9   66
4640 : 82 f0 03 4c 51 46 20 af   ac
4648 : 46 a9 00 8d df 47 4c a2   c3
4650 : 46 c9 83 f0 03 4c 75 46   6f
4658 : a9 00 a2 64 a0 02 20 a3   18
4660 : 13 20 61 49 20 5f 13 a9   a1
4668 : 7e a2 61 a0 02 20 a3 13   7a
4670 : a9 00 4c a2 46 c9 84 f0   27
4678 : 03 4c 87 46 20 c5 4a a9   f9
4680 : 00 8d df 47 4c a2 46 c9   ae
4688 : 87 f0 03 4c 96 46 20 ad   49
4690 : 13 a9 00 4c a2 46 c9 8a   9a
4698 : f0 03 4c a2 46 20 78 25   03
46a0 : a9 00 c9 1b d0 03 4c ac   cf
46a8 : 46 4c 12 46 4c d4 34 8d   b9
46b0 : df 47 20 36 24 20 5f 13   e9
46b8 : 20 6e 13 20 b9 25 20 7d   18
46c0 : ff 49 4e 53 45 52 54 20   da
46c8 : 44 49 53 4b 20 54 4f 20   11
46d0 : 42 45 20 00 ad df 47 c9   47
46d8 : 82 d0 0f 20 7d ff 46 4f   1a
46e0 : 52 4d 41 54 45 44 00 4c   c3
46e8 : f7 46 20 7d ff 56 41 4c   0a
46f0 : 49 44 41 54 45 44 00 20   ed
46f8 : 7d ff 20 49 4e 20 44 52   42
4700 : 49 56 45 20 23 20 00 ad   58
4708 : 7c 33 20 fb f9 48 8a 20   f2
4710 : d2 ff 68 20 d2 ff a2 0c   d0
4718 : a0 0a 20 c3 25 20 7d ff   87
4720 : 48 49 54 20 20 12 20 53   e0
4728 : 50 41 43 45 20 92 20 20   ea
4730 : 54 4f 20 43 4f 4e 54 49   e8
4738 : 4e 55 45 20 4f 52 20 20   cf
4740 : 12 20 20 45 53 43 20 20   23
4748 : 92 20 20 54 4f 20 4c 45   2f
4750 : 41 56 45 00 20 e4 ff c9   ca
4758 : 1b f0 04 c9 20 d0 f5 c9   1a
4760 : 1b d0 06 48 20 5f 13 68   88
4768 : 60 ad df 47 c9 82 f0 01   f6
4770 : 60 20 5f 13 20 b9 25 a9   d2
4778 : 13 8d 9e 13 20 7d ff 50   ea
4780 : 4c 45 41 53 45 20 45 4e   31
4788 : 54 45 52 20 48 45 41 44   54
4790 : 45 52 20 41 4e 44 20 49   49
4798 : 44 20 3a 20 00 a9 0c a2   42
47a0 : 28 a0 13 20 6f 29 a9 17   f6
47a8 : a0 29 20 b8 28 a2 06 a0   ed
47b0 : 00 20 2e ca 20 68 13 20   77
47b8 : 3e 2f ad 81 33 30 16 20   77
47c0 : c2 24 a9 15 a2 29 ac 14   f0
47c8 : 29 c8 c8 85 89 86 8a 8c   48
47d0 : 7b 33 20 16 30 20 8a 2f   3c
47d8 : 20 e0 47 20 78 25 60 00   70
47e0 : 20 3e 2f ad 81 33 30 10   33
47e8 : a9 fc a2 47 a0 02 85 89   e4
47f0 : 86 8a 8c 7b 33 20 16 30   3b
47f8 : 20 8a 2f 60 55 3a 20 4f   7b
4800 : 18 c9 1b d0 01 60 20 07   7f
4808 : 4e a9 06 a2 00 20 2e ca   50
4810 : 20 e4 ff c9 81 f0 03 4c   20
4818 : 87 48 20 5f 13 a2 10 8e   5b
4820 : 14 29 a9 20 9d 17 29 ca   04
4828 : 10 fa a0 90 20 98 23 a2   88
4830 : 06 a0 0a 20 c3 25 20 6e   d0
4838 : 13 20 7d ff 50 4c 45 41   ba
4840 : 53 45 20 45 4e 54 45 52   28
4848 : 20 4e 45 57 20 48 45 41   a7
4850 : 44 45 52 20 3a 20 00 a9   c7
4858 : 10 8d 9e 13 a9 0c a2 25   09
4860 : ac 9e 13 20 6f 29 a9 17   39
4868 : a0 29 20 b8 28 20 36 24   61
4870 : a0 90 20 98 23 20 5f 13   4a
4878 : 20 aa 2a a9 06 a2 00 20   63
4880 : 2e ca a9 00 4c 56 49 c9   ae
4888 : 83 f0 03 4c fa 48 20 5f   ff
4890 : 13 a2 05 8e 14 29 a9 20   79
4898 : 9d 17 29 ca 10 fa a0 a2   05
48a0 : 20 98 23 a2 06 a0 0a 20   f7
48a8 : c3 25 20 6e 13 20 7d ff   fc
48b0 : 50 4c 45 41 53 45 20 45   0a
48b8 : 4e 54 45 52 20 4e 45 57   04
48c0 : 20 49 44 45 4e 54 2e 20   bf
48c8 : 3a 20 00 a9 05 8d 9e 13   a5
48d0 : a9 0c a2 25 a0 05 20 6f   5e
48d8 : 29 a9 17 a0 29 20 b8 28   77
48e0 : 20 36 24 a0 a2 20 98 23   0c
48e8 : 20 5f 13 20 aa 2a a9 06   2f
48f0 : a2 00 20 2e ca a9 00 4c   f3
48f8 : 56 49 c9 84 f0 03 4c 14   76
4900 : 49 a9 ff 20 b7 23 20 36   a3
4908 : 24 20 5f 13 20 aa 2a a9   ca
4910 : 00 4c 56 49 c9 82 f0 03   70
4918 : 4c 2e 49 a9 00 20 b7 23   29
4920 : 20 36 24 20 5f 13 20 aa   cd
4928 : 2a a9 00 4c 56 49 c9 55   32
4930 : f0 03 4c 56 49 ad 83 33   f6
4938 : 8d 33 4e 8d 82 33 a9 00   0c
4940 : 8d 83 33 a9 12 8d 7f 33   83
4948 : 20 55 33 ad 33 4e 8d 83   78
4950 : 33 20 e0 47 a9 1b c9 1b   85
4958 : d0 03 4c 60 49 4c 10 48   91
4960 : 60 20 4f 18 c9 1b d0 03   66
4968 : 4c be 4a 20 5f 13 20 c2   3f
4970 : 24 20 44 17 ad 81 33 10   6c
4978 : 03 4c be 4a 20 2c 15 20   92
4980 : 5f 13 20 72 2b 20 7a 13   83
4988 : 20 aa 2a a9 00 8d 38 16   37
4990 : ad 38 16 8d d8 2c a9 06   32
4998 : a2 00 20 2e ca 20 5f 13   5a
49a0 : a9 03 a2 10 a0 03 20 d7   c8
49a8 : 2b 20 e4 ff c9 2b f0 03   dc
49b0 : 4c cc 49 20 b3 25 ae 38   48
49b8 : 16 e0 90 f0 05 ad b0 2c   59
49c0 : f0 ce a9 00 8d 38 16 f0   57
49c8 : c7 4c b4 4a c9 82 f0 03   a6
49d0 : 4c db 49 a2 85 20 7f 1b   3e
49d8 : 4c b4 4a c9 84 f0 03 4c   bf
49e0 : ea 49 a2 86 20 7f 1b 4c   eb
49e8 : b4 4a c9 55 f0 03 4c f9   2b
49f0 : 49 20 84 33 a9 1b 4c b4   df
49f8 : 4a c9 81 f0 03 4c 0f 4a   09
4a00 : a9 02 a2 13 20 ff 22 20   80
4a08 : b1 2c a9 00 4c b4 4a c9   61
4a10 : 83 f0 03 4c 54 4a 20 db   26
4a18 : 23 a2 00 a9 2c 9d 17 29   20
4a20 : e8 a9 55 9d 17 29 e8 8e   61
4a28 : c4 4a ad 7c 33 20 fb f9   24
4a30 : 48 8a c9 30 f0 09 ae c4   d1
4a38 : 4a 9d 17 29 ee c4 4a 68   4b
4a40 : ae c4 4a 9d 17 29 e8 a9   48
4a48 : 3a 9d 17 29 20 e6 23 a9   55
4a50 : 00 4c b4 4a c9 85 f0 03   7f
4a58 : 4c b4 4a 20 db 23 a2 0a   0a
4a60 : 8e 14 29 a2 10 a0 0a 20   05
4a68 : c3 25 20 6e 13 20 7d ff   bc
4a70 : 50 4c 45 41 53 45 20 45   ca
4a78 : 4e 54 45 52 20 46 49 4c   7e
4a80 : 45 2d 45 58 54 45 4e 53   08
4a88 : 49 4f 4e 20 3a 20 00 a9   08
4a90 : 0a 8d 9e 13 a9 16 a2 2a   95
4a98 : ac 9e 13 20 6f 29 a9 17   71
4aa0 : a0 29 20 b8 28 20 36 24   99
```

```
4aa8 : a9 06 a2 00 20 2e ca 20  dc
4ab0 : e6 23 a9 00 c9 1b d0 03  51
4ab8 : 4c be 4a 4c a9 49 20 5f  a4
4ac0 : 13 a9 00 60 00 20 af 46  00
4ac8 : c9 1b d0 01 60 20 c2 24  cd
4ad0 : a9 12 a2 00 8d 7f 33 8e  ea
4ad8 : 80 33 a9 00 a2 ea 85 fe  f2
4ae0 : 86 ff 20 3e 2f ad 81 33  03
4ae8 : 10 03 4c 4f 4b 20 30 30  4e
4af0 : 20 5f 13 ad 81 33 10 01  2e
4af8 : 60 20 2c 15 20 56 26 20  a4
4b00 : aa 2a ad 94 13 f0 0b 20  e2
4b08 : 5d 4b ad 81 33 10 03 4c  ff
4b10 : 4f 4b 20 3e 2f a9 59 a2  c0
4b18 : 4b a0 01 85 89 86 8a 8c  b4
4b20 : 7b 33 20 16 30 ad 94 13  e9
4b28 : f0 25 20 8a 2f ad 81 33  d1
4b30 : 30 1d 20 fb 1f a9 01 a2  ff
4b38 : 00 a0 30 8c 24 15 8c 25  8d
4b40 : 15 8c 26 15 20 9e 26 ad  b3
4b48 : 81 33 30 03 20 91 4b 20  cb
4b50 : 8a 2f 20 e0 47 20 5f 13  af
4b58 : 60 56 20 56 26 20 3e 2f  71
4b60 : a9 30 8d 24 15 8d 25 15  86
4b68 : 8d 26 15 20 0c 15 a2 0d  60
4b70 : 20 c9 ff a9 00 20 d2 ff  f6
4b78 : 20 d2 ff 20 cc ff a9 01  7b
4b80 : a2 00 8d 7f 33 8e 80 33  86
4b88 : a9 32 20 e2 2f 20 8a 2f  2b
4b90 : 60 20 3e 2f a9 30 8d 24  10
4b98 : 15 8d 25 15 8d 26 15 20  fe
4ba0 : 0c 15 a2 0d 20 c9 ff a9  25
4ba8 : 43 20 d2 ff a9 42 20 d2  83
4bb0 : ff 20 cc ff a9 01 a2 00  20
4bb8 : 8d 7f 33 8e 80 33 a9 32  50
4bc0 : 20 e2 2f 20 8a 2f ae 9c  37
4bc8 : 13 a9 01 20 ce 2d 60 a9  1f
4bd0 : 41 a2 65 a0 02 20 a3 13  a6
4bd8 : 20 03 13 a2 10 a9 20 20  a2
4be0 : d2 ff ca d0 f8 20 7d ff  05
4be8 : 20 20 42 20 4f 20 4f 20  20
4bf0 : 54 20 20 20 2d 20 20 20  f5
4bf8 : 48 20 45 20 4c 20 50 20  ed
4c00 : 20 20 00 a2 11 a9 20 20  a4
4c08 : d2 ff ca d0 f8 20 30 13  1e
4c10 : 20 e4 ff c9 81 f0 03 4c  20
4c18 : 22 4c 20 9c 4f a9 00 4c  d7
4c20 : 7c 4c c9 87 f0 03 4c 31  e0
4c28 : 4c 20 ad 13 a9 00 4c 7c  17
4c30 : 4c c9 8a f0 03 4c 40 4c  4e
4c38 : 20 78 25 a9 00 4c 7c 4c  00
4c40 : c9 82 f0 03 4c 4f 4c 20  98
4c48 : 89 4c a9 00 4c 7c 4c c9  cf
4c50 : 84 f0 03 4c 6b 4c 20 4f  cf
4c58 : 18 c9 1b f0 06 20 c2 24  ee
4c60 : 20 5a 4b 20 5f 13 a9 00  b9
4c68 : 4c 7c 4c c9 86 f0 03 4c  d3
4c70 : 7c 4c a9 82 20 af 46 a9  b9
4c78 : 00 8d df 47 c9 1b d0 03  de
4c80 : 4c 86 4c 4c 10 4c 4c d4  ea
4c88 : 34 a9 84 8d df 47 20 4f  bb
4c90 : 18 c9 1b d0 03 4c 84 4d  ad
4c98 : 20 c2 24 20 07 4e a9 06  bc
4ca0 : a2 00 20 2e ca 20 56 26  64
4ca8 : ad 9c 13 d0 40 20 89 13  d4
4cb0 : 20 5f 13 a2 0a a0 0a 20  a7
4cb8 : c3 25 20 6b 13 20 7d ff  ac
4cc0 : 54 48 45 52 45 20 49 53  f5
4cc8 : 20 4e 4f 20 42 4f 4f 54  6c
4cd0 : 2d 53 45 43 54 4f 52 20  aa
4cd8 : 4f 4e 20 54 48 49 53 20  3d
4ce0 : 44 49 53 4b 20 21 00 20  52
4ce8 : dc 4d 4c 84 4d ad 9c 13  e9
4cf0 : 8d 90 4d c9 10 90 46 20  31
4cf8 : 89 13 20 5f 13 a2 0a a0  af
4d00 : 0a 20 c3 25 20 6b 13 20  9a
4d08 : 7d ff 53 4f 52 52 59 2c  b9
4d10 : 20 54 4f 4f 20 4d 55 43  60
4d18 : 48 20 42 4f 4f 54 2d 42  bc
4d20 : 4c 4f 43 4b 53 20 46 4f  3c
4d28 : 52 20 54 48 49 53 20 50  f9
4d30 : 52 4f 47 52 41 4d 00 20  05
4d38 : dc 4d 4c 84 4d a9 00 a2  c6
4d40 : ea 85 fe 86 ff a9 01 a2  14
4d48 : ff 8d 7f 33 8e 80 33 e6  dc
4d50 : ff 20 8f 30 ce 9c 13 d0  09
4d58 : f6 20 34 4e c9 1b f0 24  b7
4d60 : 20 5f 13 a2 06 a0 0a 20  17
4d68 : c3 25 20 6e 13 20 7d ff  bc
4d70 : 43 4f 50 59 20 43 4f 4d  8e
4d78 : 50 4c 45 54 45 44 20 21  03
4d80 : 00 20 dc 4d 20 36 24 20  f6
4d88 : 5f 13 a9 00 8d df 47 60  91
4d90 : 00 a2 0d a0 0a 20 c3 25  34
4d98 : 20 6e 13 20 7d ff 48 49  44
4da0 : 54 20 12 20 20 53 50 41  ed
4da8 : 43 45 20 20 92 20 54 4f  b4
4db0 : 20 43 4f 4e 54 49 4e 55  83
4db8 : 45 20 4f 52 20 12 20 20  7f
4dc0 : 45 53 43 20 20 92 20 54  43
4dc8 : 4f 20 41 42 4f 52 54 00  99
4dd0 : 20 e4 ff c9 1b f0 04 c9  78
4dd8 : 20 d0 f5 60 a2 0d a0 0a  13
4de0 : 20 c3 25 20 6e 13 20 7d  2a
4de8 : ff 48 49 54 20 41 4e 59  e0
4df0 : 20 4b 45 59 20 54 4f 20  54
4df8 : 43 4f 4e 54 49 4e 55 45  e8
4e00 : 00 20 e4 ff f0 fb 60 20  fa
4e08 : 5f 13 a2 05 a0 02 20 cc  6e
4e10 : 24 a9 12 a2 00 8d 7f 33  b3
4e18 : 8e 80 33 a9 00 a2 ea 85  b4
4e20 : fe 86 ff 20 3e 2f 20 30  a3
4e28 : 30 20 2c 15 20 5f 13 20  a0
4e30 : aa 2a 60 00 20 5f 13 a2  96
4e38 : 0a a0 0a 20 c3 25 20 6e  dc
4e40 : 13 20 7d ff 20 49 4e 53  ef
4e48 : 45 52 54 20 44 45 53 54  34
4e50 : 49 4e 41 54 49 4f 4e 20  24
4e58 : 44 49 53 4b 20 49 4e 20  45
4e60 : 41 43 54 55 41 4c 20 44  82
4e68 : 52 49 56 45 20 21 00 20  e8
4e70 : 91 4d c9 1b d0 03 4c 9b  0b
4e78 : 4f 8d df 47 20 07 4e a9  35
4e80 : 09 a2 00 8e df 47 20 2e  c1
4e88 : ca 20 56 26 ad 9c 13 d0  6a
4e90 : 03 4c cc 4e 20 89 13 20  91
4e98 : 5f 13 a2 0a a0 0a 20 c3  cd
4ea0 : 25 20 6b 13 20 7d ff 41  83
4ea8 : 4c 52 45 41 44 59 20 41  a9
4eb0 : 20 42 4f 4f 54 20 2d 20  ea
4eb8 : 44 49 53 4b 20 21 00 20  2a
4ec0 : 91 4d c9 1b f0 03 4c 55  d1
4ec8 : 4f 4c 9b 4f 20 5f 13 a9  ab
4ed0 : 00 a2 ea 85 fe 86 ff ad  0c
4ed8 : 90 4d 8d 9c 13 a0 03 20  88
4ee0 : 04 25 c9 15 d0 03 4c 55  8d
4ee8 : 4f a0 04 20 04 25 a0 08  88
4ef0 : 6a 90 0c ce 9c 13 f0 5d  60
4ef8 : 88 d0 f5 a0 05 d0 ec 20  45
4f00 : 89 13 20 5f 13 a2 0a a0  b7
4f08 : 0a 20 c3 25 20 6b 13 20  a2
4f10 : 7d ff 52 45 51 55 49 45  3a
4f18 : 52 45 44 20 53 50 41 43  65
4f20 : 45 20 4f 4e 20 44 49 53  03
4f28 : 4b 20 49 53 20 4f 43 43  50
4f30 : 55 50 49 45 44 20 42 59  a9
4f38 : 20 41 4e 4f 54 48 45 52  b8
4f40 : 20 50 52 4f 47 52 41 4d  ad
4f48 : 20 21 00 20 91 4d c9 1b  de
4f50 : d0 03 4c 9b 4f 20 5f 13  c2
4f58 : a2 05 a0 05 20 cc 24 a9  92
4f60 : 00 a2 ea 85 fe 86 ff a9  94
4f68 : 01 a2 00 8d 7f 33 8e 80  39
4f70 : 33 e6 ff ad 83 33 8d 33  3a
4f78 : 4e ad 90 4d 8d 83 33 8d  47
4f80 : 9c 13 20 ce 30 ad 33 4e  62
4f88 : 8d 83 33 ad 81 33 30 0b  e2
4f90 : ae 9c 13 a9 01 20 ce 2d  2d
4f98 : 20 e0 47 60 a9 82 a2 66  0c
4fa0 : a0 02 20 a3 13 20 e4 ff  84
4fa8 : c9 83 f0 03 4c 55 50 20  c0
4fb0 : b9 25 20 6e 13 20 7d ff  fa
4fb8 : 49 4e 53 45 52 54 20 53  95
4fc0 : 4f 55 52 43 45 2d 44 49  18
4fc8 : 53 4b 20 49 4e 20 41 43  64
4fd0 : 54 55 41 4c 20 44 52 49  a9
4fd8 : 56 45 00 20 91 4d c9 1b  b6
4fe0 : f0 68 20 7f 51 ae cb 2d  10
4fe8 : f0 60 e8 8e 90 4d 20 5b  bf
4ff0 : 52 a5 16 d0 04 a5 17 f0  60
4ff8 : 51 a5 16 8d 0f 2f a5 17  82
5000 : 8d 10 2f 20 c5 51 a5 16  0f
5008 : d0 04 a5 17 f0 3c 8d 15  78
5010 : 2f a5 16 8d 1d 68 a5 17  23
5018 : 8d 1e 68 a9 4c 8d 1c 68  76
5020 : a9 00 8d a9 2a 20 0f 52  e6
5028 : 20 79 2a a9 00 a2 00 a0  1b
5030 : ea c8 c8 20 d5 ff b0 12  f9
5038 : ae 90 4d ca 8e 12 2f a9  64
5040 : 00 8d a9 2a 20 81 2e 20  be
5048 : 34 4e 20 5f 13 a9 00 8d  31
5050 : 15 2f 4c 61 51 c9 85 f0  97
5058 : 03 4c c4 50 20 4f 18 c9  2d
5060 : 1b f0 56 a2 01 8e 90 4d  3f
5068 : ca 8e 0f 2f 8e 10 2f 20  89
5070 : 0f 52 a2 00 8e a9 2a 20  70
5078 : 5f 13 a2 05 a0 0a 20 c3  0c
5080 : 25 a9 10 8d 9e 13 20 2c  8b
5088 : 2a 20 c5 51 a5 16 d0 0a  c0
5090 : a5 17 d0 06 20 89 13 4c  e9
5098 : b9 50 8d 15 2f a5 16 8d  13
50a0 : 1d 68 a5 17 8d 1e 68 a9  fc
50a8 : 4c 8d 1c 68 ae 90 4d ca  09
50b0 : 8e 12 2f 20 81 2e 20 34  8a
50b8 : 4e 20 5f 13 a9 00 8d 15  4c
50c0 : 2f 4c 61 51 c9 81 f0 03  0a
50c8 : 4c 46 51 20 4f 18 c9 1b  a3
50d0 : d0 03 4c 3e 51 a2 01 8e  48
50d8 : 90 4d ca 8e 0f 2f 8e 10  58
50e0 : 2f 20 0f 52 a2 00 8e a9  e5
50e8 : 2a 20 5f 13 a2 05 a0 0a  45
50f0 : 20 c3 25 a9 10 8d 9e 13  7e
50f8 : 20 2c 2a a2 00 8e 1c 53  99
5100 : e8 8e 5a 53 20 a4 52 ae  fe
5108 : 5a 53 f0 25 a2 00 a0 00  99
5110 : bd 78 51 9d 1c 68 e8 e0  7c
5118 : 07 90 f5 ca b9 1c 53 f0  ea
5120 : 09 e8 9d 1c 68 c8 c0 3d  d2
5128 : 90 f2 e8 9d 1c 68 8c 5a  0b
5130 : 53 ae 90 4d ca 8e 12 2f  70
5138 : 20 81 2e 20 34 4e 20 5f  9d
5140 : 13 a9 00 4c 61 51 c9 87  88
5148 : f0 03 4c 55 51 20 ad 13  6b
5150 : a9 00 4c 61 51 c9 8a f0  a8
5158 : 03 4c 61 51 20 78 25 a9  b2
5160 : 00 c9 1b d0 03 4c 6b 51  09
5168 : 4c a5 4f 20 5f 13 a9 41  16
5170 : a2 65 a0 02 20 a3 13 60  59
5178 : a2 00 a0 00 4c cd 4d 20  eb
```

Listing 1. »Power 128/v.1.1« (Fortsetzung)

```
5180 : b9 25 a9 10 8d 28 15 20   e7
5188 : 6e 13 20 7d ff 50 4c 45   76
5190 : 41 53 45 20 45 4e 54 45   73
5198 : 52 20 46 49 4c 45 2d 4c   f1
51a0 : 45 47 48 54 20 49 4e 20   eb
51a8 : 20 5b 42 4c 4f 43 4b 53   73
51b0 : 5d 20 3a 20 00 a9 0c a2   73
51b8 : 35 a0 02 20 6f 29 20 1e   bf
51c0 : 2d 20 36 24 60 20 b9 25   48
51c8 : 20 6e 13 20 7d ff 50 4c   9a
51d0 : 45 41 53 45 20 45 4e 54   41
51d8 : 45 52 20 53 54 41 52 54   fa
51e0 : 2d 41 44 52 45 53 53 20   86
51e8 : 20 28 49 4e 20 44 45 43   f8
51f0 : 49 4d 41 4c 53 20 21 29   c7
51f8 : 20 3a 20 00 a9 0c a2 37   31
5200 : a0 05 8c c9 2d 20 6f 29   63
5208 : 20 1e 2d 20 36 24 60 20   cd
5210 : db 23 20 b9 25 20 6e 13   ef
5218 : 20 7d ff 45 4e 54 45 52   e1
5220 : 20 42 4f 4f 54 2d 4d 45   8e
5228 : 53 53 41 47 45 20 49 46   65
5230 : 20 59 4f 55 20 4c 49 4b   9b
5238 : 45 00 a9 0e a2 0a a0 3c   1f
5240 : 20 6f 29 a9 17 a0 29 a2   f8
5248 : 3c 8e 9e 13 8e 14 2f 20   5c
5250 : b8 28 20 36 24 a2 00 8e   60
5258 : 14 2f 60 20 b9 25 20 6e   42
5260 : 13 20 7d ff 50 4c 45 41   e2
5268 : 53 45 20 45 4e 54 45 52   50
5270 : 20 4c 4f 41 44 2d 41 44   ee
5278 : 52 45 53 53 20 20 28 49   e2
5280 : 4e 20 44 45 43 49 4d 41   ce
5288 : 4c 53 20 21 29 20 3a 20   67
5290 : 00 a9 0c a2 37 a0 05 8c   62
5298 : c9 2d 20 6f 29 20 1e 2d   54
52a0 : 20 36 24 60 a2 3c a9 00   a3
52a8 : 9d 1c 53 ca d0 f8 20 b9   4a
52b0 : 25 20 6e 13 20 7d ff 4e   6e
52b8 : 4f 57 20 45 4e 54 45 52   a5
52c0 : 20 42 41 53 49 43 2d 43   a6
52c8 : 4f 4d 4d 41 4e 44 53 2c   e6
52d0 : 20 53 45 50 45 52 41 54   8a
52d8 : 45 44 20 42 59 20 20 5b   5e
52e0 : 3a 5d 2c 20 45 58 45 43   8b
52e8 : 55 54 45 44 20 41 46 54   0f
52f0 : 45 52 20 42 4f 4f 54 49   02
52f8 : 4e 47 00 a9 0e a2 0a a0   7e
5300 : 3c 20 6f 29 a9 1c a0 53   f2
5308 : a2 3c 8e 9e 13 8e 59 53   f1
5310 : 20 b8 28 20 36 24 a2 00   aa
5318 : 8e 59 53 60 00 00 00 00   34
5320 : 00 00 00 00 00 00 00 00   21
5328 : 00 00 00 00 00 00 00 00   29
5330 : 00 00 00 00 00 00 00 00   31
5338 : 00 00 00 00 00 00 00 00   39
5340 : 00 00 00 00 00 00 00 00   41
5348 : 00 00 00 00 00 00 00 00   49
5350 : 00 00 00 00 00 00 00 00   51
5358 : 00 00 00 a2 14 a9 10 20   bc
5360 : c2 53 8d 2f 0a a2 06 a9   36
5368 : 1e 20 c2 53 a2 07 a9 22   ff
5370 : 20 c2 53 60 8d b2 53 8e   ab
5378 : b3 53 a9 07 8d bd 53 a9   87
5380 : d0 8d be 53 88 f0 14 18   c1
5388 : ad be 53 69 50 8d be 53   a9
5390 : a9 00 6d bd 53 8d bd 53   8c
5398 : 4c 84 53 ad b2 53 d0 03   c0
53a0 : ce b3 53 ce b2 53 ee b2   dd
53a8 : 53 d0 03 ee b3 53 20 ce   f6
53b0 : 53 ad 00 00 f0 06 20 c0   1b
53b8 : 53 4c a6 53 60 00 00 c4   d5
53c0 : a2 1f 8e 00 d6 2c 00 d6   12
53c8 : 10 fb 8d 01 d6 60 a2 12   78
```

```
53d0 : 18 a9 10 6d bd 53 20 c2   eb
53d8 : 53 e8 ad be 53 20 c2 53   ca
53e0 : ad bf 53 20 c0 53 a2 12   9b
53e8 : ad bd 53 20 c2 53 e8 ad   12
53f0 : be 53 20 c2 53 d8 18 69   e7
53f8 : 01 8d be 53 90 03 ee bd   32
5400 : 53 60 20 20 20 20 20 20   53
5408 : 20 20 20 20 20 20 20 20   08
5410 : 20 20 20 20 20 20 20 20   10
5418 : 20 20 20 20 20 20 20 20   18
5420 : 20 20 20 20 20 20 20 20   20
5428 : 20 20 20 20 20 20 20 20   28
5430 : 20 20 20 20 20 20 20 20   30
5438 : 20 20 20 20 20 20 20 20   38
5440 : 20 20 20 20 20 20 20 20   40
5448 : 20 20 20 20 20 20 20 20   48
5450 : 20 20 20 20 20 20 20 20   50
5458 : 20 20 20 20 44 20 49 20   3f
5460 : 52 20 45 20 43 20 54 20   df
5468 : 4f 20 52 20 59 20 2d 20   ec
5470 : 20 20 20 20 41 20 4e 20   3b
5478 : 44 20 20 20 20 20 42 20   25
5480 : 4f 20 4f 20 54 20 2d 20   f2
5488 : 55 20 54 20 49 20 4c 20   0e
5490 : 49 20 54 20 59 20 20 20   5a
5498 : 20 20 20 20 20 20 20 20   98
54a0 : 20 20 20 20 20 20 20 20   a0
54a8 : 20 20 20 20 20 20 20 20   a8
54b0 : 20 20 20 20 46 20 4f 20   cf
54b8 : 52 20 20 20 20 54 20 48   dc
54c0 : 20 45 20 20 20 20 43 20   df
54c8 : 4f 20 4d 20 4d 20 4f 20   d2
54d0 : 44 20 4f 20 52 20 45 20   78
54d8 : 20 20 31 20 32 20 38 20   9e
54e0 : 20 20 20 20 20 20 20 20   e0
54e8 : 20 20 20 20 20 20 20 20   e8
54f0 : 20 20 20 20 20 20 20 20   f0
54f8 : 20 20 20 20 17 12 09 14   83
5500 : 14 05 0e 20 02 19 20 4a   1c
5508 : 2e 20 57 01 0c 04 08 01   3f
5510 : 05 15 13 05 0c 20 09 0e   07
5518 : 20 31 39 38 38 20 06 0f   e1
5520 : 12 20 36 34 27 05 12 20   7a
5528 : 20 2f 20 20 28 03 29 20   6b
5530 : 4d 20 26 20 54 20 20 20   22
5538 : 20 20 20 20 20 20 20 20   38
5540 : 20 20 20 20 20 20 20 20   40
5548 : 20 20 20 20 20 20 20 20   48
5550 : 20 20 20 20 20 20 20 20   50
5558 : 20 20 20 20 20 20 20 20   58
5560 : 20 20 20 20 20 20 20 20   60
5568 : 20 20 20 20 20 20 20 20   68
5570 : 20 20 20 20 20 20 20 20   70
5578 : 20 20 20 20 20 20 20 20   78
5580 : 20 20 20 20 20 20 20 20   80
5588 : 20 20 20 20 20 20 20 20   88
5590 : 20 20 00 20 20 20 49 0e   09
5598 : 13 05 12 14 20 20 20 44   41
55a0 : 09 13 0b 20 20 20 09 0e   3d
55a8 : 20 20 20 01 03 14 15 01   28
55b0 : 0c 20 20 20 44 12 09 16   fe
55b8 : 05 20 20 20 14 08 05 0e   8b
55c0 : 20 20 20 08 09 14 20 20   eb
55c8 : 20 20 20 3c 20 46 31 20   c1
55d0 : 3e 20 20 20 20 20 20 20   ee
55d8 : 20 20 20 20 20 20 20 20   d8
55e0 : 20 20 20 20 20 20 20 20   e0
55e8 : 20 20 20 20 20 20 20 20   e8
55f0 : 20 20 20 20 20 20 20 20   f0
55f8 : 20 20 20 20 20 20 20 20   f8
5600 : 20 20 20 20 20 20 20 20   00
5608 : 20 20 20 20 20 20 20 20   08
5610 : 20 20 20 20 20 20 20 20   10
5618 : 20 20 20 20 20 20 20 20   18
```

```
5620 : 20 20 20 20 20 20 20 20   20
5628 : 20 20 20 20 20 20 20 20   28
5630 : 20 20 20 20 20 20 46 37   f7
5638 : 3d 20 44 09 13 0b 20 44   4a
5640 : 12 09 16 05 20 2d 20 54   92
5648 : 0f 07 07 0c 05 20 20 20   30
5650 : 20 20 20 20 20 20 20 20   50
5658 : 20 20 20 20 20 20 20 20   58
5660 : 46 38 3d 20 4f 15 14 10   24
5668 : 15 14 20 2d 20 54 0f 07   24
5670 : 07 0c 05 20 20 20 20 20   87
5678 : 20 20 20 20 20 20 20 20   78
5680 : 20 20 20 20 20 20 48 45   6b
5688 : 4c 50 3d 20 43 42 4d 20   0b
5690 : 2d 20 44 09 12 05 03 14   7d
5698 : 0f 12 19 20 20 20 20 20   bf
56a0 : 20 20 20 20 20 20 20 20   a0
56a8 : 20 20 20 20 20 20 20 20   a8
56b0 : 45 53 43 3d 20 42 01 03   35
56b8 : 0b 20 14 0f 20 4d 01 09   3d
56c0 : 0e 20 4d 05 0e 15 05 20   b0
56c8 : 20 20 20 20 20 20 20 20   c8
56d0 : 20 20 20 00 20 20 20 08   9c
56d8 : 09 14 20 20 20 3c 20 53   03
56e0 : 50 41 43 45 20 3e 20 20   ff
56e8 : 06 0f 12 20 20 50 41 55   33
56f0 : 53 45 20 20 0f 12 20 20   34
56f8 : 14 0f 20 20 03 0f 0e 14   a9
5700 : 09 0e 15 05 20 20 20 20   ba
5708 : 20 20 20 20 20 20 20 20   08
5710 : 20 20 20 20 20 20 20 20   10
5718 : 20 20 20 20 20 20 20 20   18
5720 : 20 20 20 20 00 20 20 20   1e
5728 : 46 31 3d 20 52 05 01 04   b4
5730 : 20 44 09 12 05 03 14 0f   ce
5738 : 12 19 2d 54 12 01 03 0b   f8
5740 : 20 14 0f 20 4d 05 0d 0f   82
5748 : 12 19 20 20 20 20 20 20   b7
5750 : 20 20 46 34 3d 20 57 12   ef
5758 : 09 14 05 20 20 44 09 12   1d
5760 : 05 03 14 0f 12 19 2d 54   15
5768 : 12 01 03 0b 20 20 20 20   e1
5770 : 20 20 20 20 20 20 20 20   70
5778 : 46 33 3d 20 4c 0f 01 04   f5
5780 : 20 13 01 16 05 04 20 44   a6
5788 : 09 12 05 03 14 0f 12 19   70
5790 : 20 20 20 20 20 20 20 20   90
5798 : 20 20 20 20 20 20 20 20   98
57a0 : 20 20 46 36 3d 20 53 01   4d
57a8 : 16 05 20 44 09 12 05 03   0d
57b0 : 14 0f 12 19 20 14 0f 20   13
57b8 : 46 09 0c 05 20 20 20 20   ea
57c0 : 20 20 20 20 20 20 20 20   c0
57c8 : 46 37 3d 20 44 09 13 0b   ec
57d0 : 20 44 12 09 16 05 20 2d   1d
57d8 : 20 54 0f 07 07 0c 05 20   ec
57e0 : 20 20 20 20 20 20 20 20   e0
57e8 : 20 20 20 20 20 20 20 20   e8
57f0 : 20 20 20 20 20 20 20 20   f0
57f8 : 20 20 20 20 20 20 20 20   f8
5800 : 20 20 20 20 20 20 20 20   00
5808 : 20 20 20 20 20 20 20 20   08
5810 : 20 20 20 20 20 20 20 20   10
5818 : 48 45 4c 50 3d 20 43 42   86
5820 : 4d 20 2d 20 44 09 12 05   ac
5828 : 03 14 0f 12 19 20 20 20   8f
5830 : 20 20 20 20 20 20 20 20   30
5838 : 20 20 20 20 20 20 20 20   38
5840 : 20 20 45 53 43 3d 20 42   4f
5848 : 01 03 0b 20 14 0f 20 4d   67
5850 : 01 09 0e 20 4d 05 0e 15   bd
5858 : 05 20 20 20 20 20 20 20   3d
5860 : 20 20 20 20 20 00 20 20   5f
5868 : 20 46 31 3d 20 13 05 0c   66
```

```
5870 : 05 03 14 05 04 20 0f 0e  36
5878 : 20 20 3c 20 50 52 47 20  b0
5880 : 3e 20 20 20 20 20 20 20  9e
5888 : 20 20 20 20 20 20 20 20  88
5890 : 20 20 20 46 32 3d 20 13  45
5898 : 05 0c 05 03 14 05 04 20  ff
58a0 : 0f 0e 20 20 3c 20 55 53  83
58a8 : 52 20 3e 20 20 20 20 20  62
58b0 : 20 20 20 20 20 20 20 20  b0
58b8 : 20 46 33 3d 20 13 05 0c  37
58c0 : 05 03 14 05 04 20 0f 0e  86
58c8 : 20 20 3c 20 53 45 51 20  f0
58d0 : 3e 20 20 20 20 20 20 20  ee
58d8 : 20 20 20 20 20 20 20 20  d8
58e0 : 20 20 20 46 34 3d 20 13  b5
58e8 : 05 0c 05 03 14 05 04 20  4f
58f0 : 0f 0e 20 20 3c 20 52 45  ab
58f8 : 4c 20 3e 20 20 20 20 20  ac
5900 : 20 20 20 20 20 20 20 20  00
5908 : 20 46 37 3d 20 44 09 13  2f
5910 : 0b 20 44 12 09 16 05 20  14
5918 : 2d 20 54 0f 07 07 0c 05  2f
5920 : 20 20 20 20 20 20 20 20  20
5928 : 20 20 20 20 20 20 20 20  28
5930 : 20 20 20 46 38 3d 20 4f  be
5938 : 15 14 10 15 14 20 2d 20  35
5940 : 54 0f 07 07 0c 05 20 20  68
5948 : 20 20 20 20 20 20 20 20  48
5950 : 20 20 20 20 20 20 20 20  50
5958 : 20 48 45 4c 50 3d 20 43  6d
5960 : 42 4d 20 2d 20 44 09 12  63
5968 : 05 03 14 0f 12 19 20 20  81
5970 : 20 20 20 20 20 20 20 20  70
5978 : 20 20 20 20 20 20 20 20  78
5980 : 20 20 20 45 53 43 3d 20  e6
5988 : 42 01 03 0b 20 14 0f 20  8c
5990 : 4d 01 09 0e 20 4d 05 0e  ff
5998 : 15 05 20 20 20 20 20 20  00
59a0 : 20 20 20 20 20 20 00 20  20
59a8 : 20 20 46 31 3d 20 10 12  c9
59b0 : 0f 14 05 03 14 20 44 09  d1
59b8 : 13 0b 20 20 20 20 20 20  21
59c0 : 20 20 20 20 20 20 20 20  c0
59c8 : 20 20 20 20 20 20 20 20  c8
59d0 : 20 20 20 20 46 32 3d 20  38
59d8 : 12 05 0c 05 01 13 05 20  0e
59e0 : 44 09 13 0b 20 20 20 20  93
59e8 : 20 20 20 20 20 20 20 20  e8
59f0 : 20 20 20 20 20 20 20 20  f0
59f8 : 20 20 46 33 3d 20 10 12  5a
5a00 : 0f 14 05 03 14 20 46 09  29
5a08 : 0c 05 13 20 20 20 20 20  24
5a10 : 20 20 20 20 20 20 20 20  10
5a18 : 20 20 20 20 20 20 20 20  18
5a20 : 20 20 20 20 46 34 3d 20  98
5a28 : 12 05 0c 05 01 13 05 20  5e
5a30 : 46 09 0c 05 13 20 20 20  92
5a38 : 20 20 20 20 20 20 20 20  38
5a40 : 20 20 20 20 20 20 20 20  40
5a48 : 20 20 46 37 3d 20 44 09  e9
5a50 : 13 0b 20 44 12 09 16 05  45
5a58 : 20 2d 20 54 0f 07 07 0c  ff
5a60 : 05 20 20 20 20 20 20 20  45
5a68 : 20 20 20 20 20 20 20 20  68
5a70 : 20 20 20 20 20 20 20 20  70
5a78 : 20 20 20 20 20 20 20 20  78
5a80 : 20 20 20 20 20 20 20 20  80
5a88 : 20 20 20 20 20 20 20 20  88
5a90 : 20 20 20 20 20 20 20 20  90
5a98 : 20 20 48 45 4c 50 3d 20  00
5aa0 : 43 42 4d 20 2d 20 44 09  53
5aa8 : 12 05 03 14 0f 12 19 20  a6
5ab0 : 20 20 20 20 20 20 20 20  b0
5ab8 : 20 20 20 20 20 20 20 20  b8
5ac0 : 20 20 20 20 45 53 43 3d  73
5ac8 : 20 42 01 03 0b 20 14 0f  ca
5ad0 : 20 4d 01 09 0e 20 4d 05  19
5ad8 : 0e 15 05 20 20 20 20 20  7a
5ae0 : 20 20 20 20 20 20 20 00  a0
5ae8 : 20 20 20 46 33 3d 20 10  a7
5af0 : 12 0f 14 05 03 14 20 46  0d
5af8 : 09 0c 05 13 20 20 20 20  6f
5b00 : 20 20 20 20 20 20 20 20  00
5b08 : 20 20 20 20 20 20 20 20  08
5b10 : 20 20 20 20 20 46 34 3d  cc
5b18 : 20 12 05 0c 05 01 13 05  b3
5b20 : 20 46 09 0c 05 13 20 20  d1
5b28 : 20 20 20 20 20 20 20 20  28
5b30 : 20 20 20 20 20 20 20 20  30
5b38 : 20 20 20 55 20 3d 20 55  32
5b40 : 10 04 01 14 05 20 44 09  8a
5b48 : 13 0b 20 20 20 20 20 20  b1
5b50 : 20 20 20 20 20 20 20 20  50
5b58 : 20 20 20 20 20 20 20 20  58
5b60 : 20 20 20 20 20 2b 20 3d  f3
5b68 : 20 13 08 0f 17 20 0e 05  ab
5b70 : 18 14 20 50 01 07 05 20  41
5b78 : 20 20 20 20 20 20 20 20  78
5b80 : 20 20 20 20 20 20 20 20  80
5b88 : 20 20 20 20 20 20 20 20  88
5b90 : 20 20 20 20 20 20 20 20  90
5b98 : 20 20 20 20 20 20 20 20  98
5ba0 : 20 20 20 20 20 20 20 20  a0
5ba8 : 20 20 20 20 20 20 20 20  a8
5bb0 : 20 20 20 20 20 20 20 20  b0
5bb8 : 20 20 20 20 20 20 20 20  b8
5bc0 : 20 20 20 20 20 20 20 20  c0
5bc8 : 20 20 20 20 20 20 20 20  c8
5bd0 : 20 20 20 20 20 20 20 20  d0
5bd8 : 20 20 20 20 20 20 20 20  d8
5be0 : 20 20 20 20 20 20 20 20  e0
5be8 : 20 20 20 20 20 20 20 20  e8
5bf0 : 20 20 20 20 20 20 20 20  f0
5bf8 : 20 20 20 20 20 20 20 20  f8
5c00 : 20 20 20 20 20 45 53 43  3c
5c08 : 3d 20 42 01 03 0b 20 14  37
5c10 : 0f 20 0c 01 13 14 20 4d  40
5c18 : 05 0e 15 05 20 20 20 20  ce
5c20 : 20 20 20 20 20 20 20 20  20
5c28 : 00 20 20 20 48 09 14 20  a2
5c30 : 20 20 3c 20 53 50 41 43  b7
5c38 : 45 20 3e 20 20 20 14 0f  92
5c40 : 20 20 13 14 01 12 14 20  e9
5c48 : 20 13 0f 12 14 09 0e 07  c8
5c50 : 20 20 20 20 20 20 20 20  50
5c58 : 20 20 20 20 20 20 20 20  58
5c60 : 20 20 20 20 20 20 20 20  60
5c68 : 20 20 20 20 20 20 20 20  68
5c70 : 20 20 20 20 20 20 20 20  70
5c78 : 20 20 20 20 20 20 20 20  78
5c80 : 20 20 20 20 20 20 20 20  80
5c88 : 20 20 20 20 20 20 20 20  88
5c90 : 20 20 20 20 20 20 20 20  90
5c98 : 20 20 20 20 20 20 20 20  98
5ca0 : 20 20 20 20 20 20 20 20  a0
5ca8 : 20 20 20 20 20 20 20 20  a8
5cb0 : 20 20 20 20 20 20 20 20  b0
5cb8 : 20 20 20 20 20 20 20 20  b8
5cc0 : 20 20 20 20 20 20 20 20  c0
5cc8 : 20 20 20 20 46 37 3d 20  58
5cd0 : 44 09 13 0b 20 44 12 09  3e
5cd8 : 16 05 20 2d 20 54 0f 07  0e
5ce0 : 07 0c 05 20 20 20 20 20  f7
5ce8 : 20 20 20 20 20 20 20 20  e8
5cf0 : 20 20 20 20 20 20 20 20  f0
5cf8 : 20 20 20 20 20 20 20 20  f8
5d00 : 20 20 20 20 20 20 20 20  00
5d08 : 20 20 20 20 20 20 20 20  08
5d10 : 20 20 20 20 20 20 20 20  10
5d18 : 20 20 20 20 48 45 4c 50  d5
5d20 : 3d 20 43 42 4d 20 2d 20  51
5d28 : 44 09 12 05 03 14 0f 12  47
5d30 : 19 20 20 20 20 20 20 20  29
5d38 : 20 20 20 20 20 20 20 20  38
5d40 : 20 20 20 20 20 20 45 53  3b
5d48 : 43 3d 20 42 01 03 0b 20  0f
5d50 : 14 0f 20 4d 01 09 0e 20  6e
5d58 : 4d 05 0e 15 05 20 20 20  60
5d60 : 20 20 20 20 20 20 20 20  60
5d68 : 20 00 20 20 20 46 31 3d  08
5d70 : 20 0d 0f 16 05 20 06 09  19
5d78 : 0c 05 28 13 29 20 20 20  c8
5d80 : 20 20 20 20 20 20 20 20  80
5d88 : 20 20 20 20 20 20 20 20  88
5d90 : 20 20 20 20 20 20 20 46  dd
5d98 : 32 3d 20 12 05 0d 0f 16  d4
5da0 : 05 20 44 09 12 05 03 14  65
5da8 : 0f 12 19 2d 45 0e 14 12  e6
5db0 : 19 20 20 20 20 20 20 20  a9
5db8 : 20 20 20 20 20 46 33 3d  70
5dc0 : 20 09 0e 13 05 12 14 20  bc
5dc8 : 52 0f 17 20 20 20 20 20  30
5dd0 : 20 20 20 20 20 20 20 20  d0
5dd8 : 20 20 20 20 20 20 20 20  d8
5de0 : 20 20 20 20 20 20 20 2b  f6
5de8 : 20 3d 20 13 08 0f 17 20  a7
5df0 : 0e 05 18 14 20 50 01 07  a0
5df8 : 05 20 20 20 20 20 20 20  dd
5e00 : 20 20 20 20 20 20 20 20  00
5e08 : 20 20 20 20 20 46 35 3d  c8
5e10 : 20 13 0f 12 14 20 0f 0e  5b
5e18 : 20 46 09 0c 05 2d 4e 01  14
5e20 : 0d 05 20 20 20 20 20 20  80
5e28 : 20 20 20 20 20 20 20 20  28
5e30 : 20 20 20 20 20 20 20 46  7d
5e38 : 36 3d 20 13 0f 12 14 20  89
5e40 : 0f 0e 20 46 09 0c 05 2d  87
5e48 : 4c 05 07 08 14 20 20 20  dd
5e50 : 20 20 20 20 20 20 20 20  50
5e58 : 20 20 20 20 20 55 20 3d  3c
5e60 : 20 55 10 04 01 14 05 20  b5
5e68 : 44 09 13 0b 20 20 20 20  1b
5e70 : 20 20 20 20 20 20 20 20  70
5e78 : 20 20 20 20 20 20 20 20  78
5e80 : 20 20 20 20 20 20 20 45  cb
5e88 : 53 43 3d 20 42 01 03 0b  1f
5e90 : 20 14 0f 20 0c 01 13 14  bf
5e98 : 20 4d 05 0e 15 05 20 20  9c
5ea0 : 20 20 20 20 20 20 20 20  a0
5ea8 : 20 20 00 20 20 20 48 09  13
5eb0 : 14 20 20 20 3c 20 53 50  93
5eb8 : 41 43 45 20 3e 20 20 20  96
5ec0 : 14 0f 20 20 13 14 01 12  62
5ec8 : 14 20 20 46 09 0c 05 20  03
5ed0 : 2d 20 53 05 0c 05 03 14  a0
5ed8 : 09 0f 0e 20 20 20 20 20  b4
5ee0 : 20 20 20 20 20 20 20 20  e0
5ee8 : 20 20 20 20 20 20 20 20  e8
5ef0 : 20 20 20 20 20 20 20 20  f0
5ef8 : 20 20 20 20 20 20 20 20  f8
5f00 : 20 20 20 20 20 20 20 20  00
5f08 : 20 20 20 20 20 20 20 20  08
5f10 : 20 20 20 20 20 20 20 20  10
5f18 : 20 20 20 20 20 20 20 20  18
5f20 : 20 20 20 20 20 20 20 20  20
5f28 : 20 20 20 20 20 20 20 20  28
5f30 : 20 20 20 20 20 20 20 20  30
5f38 : 20 20 20 20 20 20 20 20  38
5f40 : 20 20 20 20 20 20 20 20  40
```

Listing 1. »Power 128/v.1.1« (Fortsetzung)

```
5f48 : 20 20 20 20 20 20 46 37  0f
5f50 : 3d 20 44 09 13 0b 20 44  62
5f58 : 12 09 16 05 20 2d 20 54  aa
5f60 : 0f 07 07 0c 05 20 20 20  48
5f68 : 20 20 20 20 20 20 20 20  68
5f70 : 20 20 20 20 20 20 20 20  70
5f78 : 20 20 20 20 20 20 20 20  78
5f80 : 20 20 20 20 20 20 20 20  80
5f88 : 20 20 20 20 20 20 20 20  88
5f90 : 20 20 20 20 20 20 20 20  90
5f98 : 20 20 20 20 20 20 48 45  83
5fa0 : 4c 50 3d 20 43 42 4d 20  23
5fa8 : 2d 20 44 09 12 05 03 14  95
5fb0 : 0f 12 19 20 20 20 20 20  d7
5fb8 : 20 20 20 20 20 20 20 20  b8
5fc0 : 20 20 20 20 20 20 20 20  c0
5fc8 : 45 53 43 3d 20 42 01 03  4d
5fd0 : 0b 20 14 0f 20 4d 01 09  55
5fd8 : 0e 20 4d 05 0e 15 05 20  c8
5fe0 : 20 20 20 20 20 20 20 20  e0
5fe8 : 20 20 20 00 20 20 20 46  31
5ff0 : 31 3d 20 0d 0f 16 05 20  60
5ff8 : 06 09 0c 05 28 13 29 20  27
6000 : 20 20 20 20 20 20 20 20  00
6008 : 20 20 20 20 20 20 20 20  08
6010 : 20 20 20 20 20 20 20 20  10
6018 : 20 46 32 3d 20 12 05 0d  50
6020 : 0f 16 05 20 44 09 12 05  5f
6028 : 03 14 0f 12 19 2d 45 0e  68
6030 : 14 12 19 20 20 20 20 20  5c
6038 : 20 20 20 20 20 20 20 46  85
6040 : 33 3d 20 09 0e 13 05 12  ed
6048 : 14 20 52 0f 17 20 20 20  16
6050 : 20 20 20 20 20 20 20 20  50
6058 : 20 20 20 20 20 20 20 20  58
6060 : 20 20 20 20 20 20 20 20  60
6068 : 20 2b 20 3d 20 13 08 0f  a6
6070 : 17 20 0e 05 18 14 20 50  ff
6078 : 01 07 05 20 20 20 20 20  06
6080 : 20 20 20 20 20 20 20 20  80
6088 : 20 20 20 20 20 20 20 46  d5
6090 : 35 3d 20 01 04 04 20 52  12
6098 : 05 0d 01 12 0b 2d 4c 09  04
60a0 : 0e 05 20 02 05 0c 0f 17  94
60a8 : 20 48 05 01 04 0c 09 0e  2f
60b0 : 05 20 20 20 20 20 20 20  95
60b8 : 20 46 38 3d 20 13 05 14  88
60c0 : 20 45 0e 04 20 0f 06 20  5a
60c8 : 46 09 0c 05 2d 4c 09 13  b6
60d0 : 14 20 20 20 20 20 20 20  c4
60d8 : 20 20 20 20 20 20 20 46  25
60e0 : 37 3d 20 13 08 0f 17 20  b6
60e8 : 01 03 14 15 01 0c 20 52  a8
60f0 : 05 13 15 0c 14 20 20 20  49
60f8 : 50 20 3d 20 50 52 49 4e  05
6100 : 54 20 20 20 20 20 20 20  34
6108 : 20 45 53 43 3d 20 42 01  e8
6110 : 03 0b 20 14 0f 20 0c 01  48
6118 : 13 14 20 4d 05 0e 15 05  06
6120 : 20 20 20 20 20 20 20 20  20
6128 : 20 20 20 20 00 20 20 20  26
6130 : 46 35 3d 20 01 04 04 20  e5
6138 : 52 05 0d 01 12 0b 2d 4c  37
6140 : 09 0e 05 20 02 05 0c 0f  2c
6148 : 17 20 48 05 01 04 0c 09  95
6150 : 0e 05 20 20 20 20 20 20  b1
6158 : 20 20 46 38 3d 20 03 12  26
6160 : 05 01 14 05 20 20 4c 01  c2
6168 : 02 05 0c 20 2d 20 48 05  f3
6170 : 01 04 05 12 20 20 20 20  bb
6178 : 20 20 20 20 20 00 20 20  77
6180 : 20 46 31 3d 20 0d 0f 04  66
6188 : 09 06 19 20 20 42 41 4d  92
6190 : 20 20 20 20 20 20 20 20  90
6198 : 20 20 20 20 20 20 20 20  98
61a0 : 20 20 20 20 20 20 20 20  a0
61a8 : 20 20 20 46 32 3d 20 06  43
61b0 : 0f 12 0d 01 14 20 44 09  91
61b8 : 13 0b 20 09 0e 20 01 03  66
61c0 : 14 15 01 0c 20 44 12 09  9f
61c8 : 16 05 20 20 20 20 20 20  31
61d0 : 20 46 33 3d 20 0d 0f 04  37
61d8 : 09 06 19 20 20 44 09 12  9b
61e0 : 05 03 14 0f 12 19 2d 45  77
61e8 : 0e 14 12 09 05 13 20 20  50
61f0 : 20 20 20 20 20 20 20 20  f0
61f8 : 20 20 20 46 34 3d 20 16  d3
6200 : 01 0c 09 04 01 14 05 20  cf
6208 : 44 09 13 0b 20 09 0e 20  ba
6210 : 01 03 14 15 01 0c 20 44  b4
6218 : 12 09 16 05 20 20 20 20  99
6220 : 20 46 37 3d 20 44 09 13  47
6228 : 0b 20 44 12 09 16 05 20  2c
6230 : 2d 20 54 0f 07 07 0c 05  47
6238 : 20 20 20 20 20 20 20 20  38
6240 : 20 20 20 20 20 20 20 20  40
6248 : 20 20 20 20 20 20 20 20  48
6250 : 20 20 20 20 20 20 20 20  50
6258 : 20 20 20 20 20 20 20 20  58
6260 : 20 20 20 20 20 20 20 20  60
6268 : 20 20 20 20 20 20 20 20  68
6270 : 20 48 45 4c 50 3d 20 43  85
6278 : 42 4d 20 2d 20 44 09 12  7b
6280 : 05 03 14 0f 12 19 20 20  99
6288 : 20 20 20 20 20 20 20 20  88
6290 : 20 20 20 20 20 20 20 20  90
6298 : 20 20 20 45 53 43 3d 20  fe
62a0 : 02 01 03 0b 20 14 0f 20  64
62a8 : 4d 01 09 0e 20 4d 05 0e  17
62b0 : 15 05 20 20 20 20 20 20  18
62b8 : 20 20 20 20 20 20 00 20  38
62c0 : 20 20 46 31 3d 20 03 08  99
62c8 : 01 0e 07 05 20 44 09 13  a1
62d0 : 0b 2d 48 05 01 04 05 12  8d
62d8 : 20 20 20 20 20 20 20 20  d8
62e0 : 20 20 20 20 20 20 20 20  e0
62e8 : 20 20 20 20 46 32 3d 20  50
62f0 : 13 05 14 20 20 1b 20 42  6f
62f8 : 0c 0f 03 0b 13 20 06 12  1d
6300 : 05 05 20 1d 20 14 0f 20  53
6308 : 20 30 20 20 20 20 20 20  10
6310 : 20 20 46 33 3d 20 03 08  29
6318 : 01 0e 07 05 20 44 09 13  f1
6320 : 0b 2d 49 04 05 0e 14 09  b8
6328 : 14 19 20 20 20 20 20 20  99
6330 : 20 20 20 20 20 20 20 20  30
6338 : 20 20 20 20 46 34 3d 20  b0
6340 : 13 05 14 20 20 1b 20 42  bf
6348 : 0c 0f 03 0b 13 20 06 12  6d
6350 : 05 05 20 1d 20 14 0f 20  a3
6358 : 20 18 18 18 18 18 20 20  91
6360 : 20 20 20 20 20 20 20 20  60
6368 : 20 20 20 20 20 20 20 20  68
6370 : 20 20 20 20 20 20 20 20  70
6378 : 20 20 20 20 20 20 20 20  78
6380 : 20 20 20 20 20 20 20 20  80
6388 : 20 20 20 20 20 20 20 20  88
6390 : 20 20 20 20 20 20 20 20  90
6398 : 20 20 20 20 20 20 20 20  98
63a0 : 20 20 20 20 20 20 20 20  a0
63a8 : 20 20 20 20 20 20 20 20  a8
63b0 : 20 20 55 20 3d 20 55 10  84
63b8 : 04 01 14 05 20 44 09 13  51
63c0 : 0b 20 20 20 20 20 20 20  ab
63c8 : 20 20 20 20 20 20 20 20  c8
63d0 : 20 20 20 20 20 20 20 20  d0
63d8 : 20 20 20 20 45 53 43 3d  8b
63e0 : 20 42 01 03 0b 20 14 0f  e2
63e8 : 20 0c 01 13 14 20 4d 05  33
63f0 : 0e 15 05 20 20 20 20 20  92
63f8 : 20 20 20 20 20 20 20 00  b8
6400 : 20 20 20 46 31 3d 20 13  a5
6408 : 0f 12 14 20 44 09 12 05  08
6410 : 03 14 0f 12 19 20 02 19  f0
6418 : 20 46 09 0c 05 20 4e 01  ac
6420 : 0d 05 20 20 20 20 20 20  80
6428 : 20 20 20 20 20 46 32 3d  dc
6430 : 20 13 05 14 20 46 09 0c  0e
6438 : 05 2d 4c 05 0e 07 08 14  e9
6440 : 20 14 0f 20 20 30 20 20  77
6448 : 20 20 20 20 20 20 20 20  48
6450 : 20 20 20 46 33 3d 20 01  f1
6458 : 04 04 20 20 1b 20 2c 55  79
6460 : 18 3a 20 1d 20 20 14 0f  b3
6468 : 20 46 09 0c 05 20 4e 01  fc
6470 : 0d 05 20 20 20 20 20 20  d0
6478 : 20 20 20 20 20 46 34 3d  34
6480 : 20 13 05 14 20 46 09 0c  5e
6488 : 05 2d 4c 05 0e 07 08 14  39
6490 : 20 14 0f 20 20 36 35 35  75
6498 : 33 35 20 20 20 20 20 20  36
64a0 : 20 20 20 46 35 3d 20 01  61
64a8 : 04 04 20 20 1b 20 18 18  fe
64b0 : 18 3a 20 1d 20 20 14 0f  03
64b8 : 20 46 09 0c 05 20 4e 01  4c
64c0 : 0d 05 20 20 20 20 20 20  20
64c8 : 20 20 20 20 20 2b 20 3d  5b
64d0 : 20 13 08 0f 17 20 0e 05  13
64d8 : 18 14 20 50 01 07 05 20  a9
64e0 : 20 20 20 20 20 20 20 20  e0
64e8 : 20 20 20 20 20 20 20 20  e8
64f0 : 20 20 20 55 20 3d 20 55  ea
64f8 : 10 04 01 14 05 20 44 09  42
6500 : 13 0b 20 20 20 20 20 20  69
6508 : 20 20 20 20 20 20 20 20  08
6510 : 20 20 20 20 20 20 20 20  10
6518 : 20 20 20 20 20 45 53 43  54
6520 : 3d 20 42 01 03 0b 20 14  4f
6528 : 0f 20 0c 01 13 14 20 4d  58
6530 : 05 0e 15 05 20 20 20 20  e6
6538 : 20 20 20 20 20 20 20 20  38
6540 : 00 20 20 20 46 31 3d 20  80
6548 : 03 12 05 01 14 05 20 42  24
6550 : 0f 0f 14 2d 53 05 03 14  23
6558 : 0f 12 20 20 1b 20 31 32  58
6560 : 38 2d 4d 0f 04 05 20 1d  87
6568 : 20 20 20 20 20 20 46 32  25
6570 : 3d 20 03 0f 10 19 20 20  eb
6578 : 20 42 0f 0f 14 2d 53 05  61
6580 : 03 14 0f 12 20 20 20 20  57
6588 : 20 20 20 20 20 20 20 20  88
6590 : 20 20 20 20 20 20 20 20  90
6598 : 20 20 20 20 20 20 20 20  98
65a0 : 20 20 20 20 20 20 20 20  a0
65a8 : 20 20 20 20 20 20 20 20  a8
65b0 : 20 20 20 20 20 20 20 20  b0
65b8 : 20 20 20 20 20 20 46 34  79
65c0 : 3d 20 04 05 0c 05 14 05  f2
65c8 : 20 42 0f 0f 14 2d 53 05  b1
65d0 : 03 14 0f 12 20 20 20 20  a7
65d8 : 20 20 20 20 20 20 20 20  d8
65e0 : 20 20 20 20 46 37 3d 20  70
65e8 : 44 09 13 0b 20 44 12 09  56
65f0 : 16 05 20 2d 20 54 0f 07  26
65f8 : 07 0c 05 20 20 20 20 20  0f
6600 : 20 20 20 20 20 20 20 20  00
6608 : 20 20 20 20 20 20 46 36  cd
6610 : 3d 20 06 0f 12 0d 01 14  77
6618 : 20 44 09 13 0b 20 09 0e  f1
6620 : 20 01 03 14 15 01 0c 20  ce
6628 : 44 12 09 16 05 20 20 20  8d
6630 : 20 20 20 20 48 45 4c 50  ed
```

```
6638 : 3d 20 43 42 4d 20 2d 20  69
6640 : 44 09 12 05 03 14 0f 12  5f
6648 : 19 20 20 20 20 20 20 20  41
6650 : 20 20 20 20 20 20 20 20  50
6658 : 20 20 20 20 20 20 45 53  53
6660 : 43 3d 20 42 01 03 0b 20  27
6668 : 14 0f 20 4d 01 09 0e 20  86
6670 : 4d 05 0e 15 05 20 20 20  78
6678 : 20 20 20 20 20 20 20 20  78
6680 : 20 00 20 20 20 46 31 3d  20
6688 : 20 0c 0f 01 04 20 01 0e  f4
6690 : 19 20 0b 09 0e 04 20 0f  3d
6698 : 06 20 50 12 0f 07 12 01  78
66a0 : 0d 20 01 0e 04 20 05 18  45
66a8 : 05 03 15 14 05 20 43 0f  73
66b0 : 0d 0d 01 0e 04 13 20 01  a1
66b8 : 06 14 05 12 20 02 0f 0f  b8
66c0 : 14 09 0e 07 20 20 20 20  81
66c8 : 20 20 20 20 20 20 20 20  c8
66d0 : 20 20 20 20 20 46 33 3d  88
66d8 : 20 01 04 04 20 46 09 0c  6b
66e0 : 05 20 14 0f 20 42 0f 0f  4b
66e8 : 14 2d 53 05 03 14 0f 12  3a
66f0 : 28 13 29 20 20 20 20 20  b4
66f8 : 20 20 20 20 20 20 20 46  45
6700 : 35 3d 20 02 0f 0f 14 20  16
6708 : 01 20 4d 41 43 48 49 4e  cd
6710 : 45 20 2d 20 50 12 0f 07  95
6718 : 12 01 0d 20 20 20 20 20  b6
6720 : 20 20 20 20 20 46 37 3d  e8
6728 : 20 44 09 13 0b 20 44 12  f6
6730 : 09 16 05 20 2d 20 54 0f  cd
6738 : 07 07 0c 05 20 20 20 20  2a
6740 : 20 20 20 20 20 20 20 20  40
6748 : 20 20 20 20 20 20 20 20  48
6750 : 20 20 20 20 20 20 20 20  50
6758 : 20 20 20 20 20 20 20 20  58
6760 : 20 20 20 20 20 20 20 20  60
6768 : 20 20 20 20 20 20 20 20  68
6770 : 20 20 20 20 20 48 45 4c  9e
6778 : 50 3d 20 43 42 4d 20 2d  41
6780 : 20 44 09 12 05 03 14 0f  1e
6788 : 12 19 20 20 20 20 20 20  f7
6790 : 20 20 20 20 20 20 20 20  90
6798 : 20 20 20 20 20 20 20 45  e3
67a0 : 53 43 3d 20 42 01 03 0b  37
67a8 : 20 14 0f 20 0c 01 13 14  d7
67b0 : 20 4d 05 0e 15 05 20 20  b4
67b8 : 20 20 20 20 20 20 20 20  b8
67c0 : 20 20 00 20 20 20 20 20  b8
67c8 : 20 20 20 20 20 20 20 20  c8
67d0 : 20 20 20 20 20 20 52 45  e3
67d8 : 4d 41 52 4b 20 3a 20 20  58
67e0 : 20 20 20 20 20 20 20 20  e0
67e8 : 20 20 20 20 20 20 20 20  e8
67f0 : 20 20 20 20 20 20 20 20  f0
67f8 : 20 20 20 20 20 20 20 20  f8
6800 : 20 20 20 20 20 20 20 20  00
6808 : 20 20 20 20 20 20 20 20  08
6810 : 20 20 20 20 20 20 20 20  10
6818 : 20 20 20 20 60 60 60 60  a0
6820 : 60 60 60 60 60 60 60 60  20
6828 : 60 60 60 60 60 60 60 60  28
6830 : 60 60 60 60 60 60 60 60  30
6838 : 60 60 60 60 60 60 60 60  38
6840 : 60 60 60 60 60 60 60 60  40
6848 : 60 60 60 60 60 60 60 60  48
6850 : 60 60 60 60 60 60 60 60  50
6858 : 60 60 60 60 60 60 60 60  58
6860 : 60 60 60 60 60 60 60 60  60
6868 : 60 60 60 60 60 60 60 60  68
6870 : 60 60 60 60 60 60 60 60  70
6878 : 60 60 60 60 60 60 60 60  78
6880 : 60 60 60 60 60 60 60 60  80
6888 : 60 60 60 60 60 60 60 60  88
6890 : 60 60 60 60 60 60 60 60  90
6898 : 60 60 60 60 60 60 60 60  98
68a0 : 60 60 60 60 60 60 60 60  a0
68a8 : 60 60 60 60 60 60 60 60  a8
68b0 : 60 60 60 60 60 60 60 60  b0
68b8 : 60 60 60 60 60 60 60 60  b8
68c0 : 60 60 60 60 60 60 60 60  c0
68c8 : 60 60 60 60 60 60 60 60  c8
68d0 : 60 60 60 60 60 60 60 60  d0
68d8 : 60 60 60 60 60 60 60 60  d8
68e0 : 60 60 60 60 60 60 60 60  e0
68e8 : 60 60 60 60 60 60 60 60  e8
68f0 : 60 60 60 60 60 60 60 60  f0
68f8 : 60 60 60 60 60 60 60 60  f8
6900 : 60 60 60 60 60 60 60 60  00
6908 : 60 60 60 60 60 60 60 60  08
6910 : 60 60 60 60 60 60 00 00  ce
6918 : 00 ff 00 ff 00 ff 00 ff  18
```

Listing 1. »Power 128/v.1.1« (Schluß)

Phantastische Schriften mit dem NEC-P 2200

Für viele 9-Nadel-Drucker gibt es bereits Zeicheneditoren, die erstaunliche Schriftqualitäten beim Ausdruck beweisen. Aber warum nicht auch die Qualität eines 24-Nadel-Druckers wie den NEC-P 2200 nutzen? Genau das erreicht unser Programm »Zenec v2.0«. Damit wird es ausgesprochen leicht, eigene Zeichen zu generieren und diese mit dem Drucker auszugeben.

Ihrer Kreativität sind beim Editieren neuer Zeichen fast keine Grenzen gesetzt: Ein Editierfeld von nicht weniger als 24 Zeilen und 37 Spalten reicht aus, um eine Vielzahl unterschiedlicher Zeichen zu erstellen.

Sie können Zeichensätze vier verschiedener Matrix-Typen editieren, auf Diskette speichern, von Diskette laden und natürlich auch zum Drucker schicken. Kleine Hilfen wie Bewegen der Zeichen in der Matrix, wenden und kopieren auf andere Zeichen sind für ein Programm dieser Güte selbstverständlich.

Die neuen Zeichensätze können problemlos zum Drucker geschickt werden, wo sie anschließend für die Anwendung unter einer Textverarbeitung (z.B. Mastertext 128) zur Verfügung stehen.

Als Hardware-Voraussetzungen benötigen Sie den C 128 im 80-Zeichen-Modus und einen NEC-P 2200, den Sie mit

einem geeigneten Kabel am Userport des Computers anschließen. Und natürlich benötigen Sie nach dem Start des Programms etwas Kreativität, um Schriftzeichen zu editieren.

So arbeiten Sie mit dem Programm

Geben Sie zuerst Listing 1 mit dem MSE ein. Eingabehinweise dazu finden Sie auf Seite 158. Nach dem Speichern auf Diskette wird das Programm mit

```
RUN"ZENEC V2.0"
```

gestartet.

Achtung: Die ASCII/DIN-Taste darf beim Start nicht gedrückt sein. Andernfalls fordert das Programm Sie vorher auf, die Taste zu lösen.

Im ersten Eingangsmenü (Bild 1) entscheiden Sie zunächst, welche der vier Matrix-Typen Sie bearbeiten wollen. Diese wählen Sie über die Tasten <A> bis <D> aus. Anschließend stehen die verschiedenen Funktionen des Hauptmenüs zur Verfügung:

<F1> Edit

Nach Betätigung der Funktionstaste <F1> gelangen Sie in den Zeichengenerator (Bild 2), dessen Funktionen weiter unten ausführlicher erläutert werden.

<F3> Load

Mit der Funktionstaste <F3> laden Sie einen bereits erstellten Zeichensatz von Diskette. Dabei können Sie diese komfortabel aus dem Directory heraus laden. Das Directory wird übersichtlich auf dem Bildschirm dargestellt. Mit den Cursortasten wird es auf- und abgescrollt. Der Dateiname unter dem reversen Balken wird durch Drücken mit der RETURN-Taste geladen.

Ein Dateiname für einen Zeichensatz muß nicht mit einem Punkt und dem Schrifttyp enden. Ist das 16. Zeichen ein A, B, C oder D, versteht es das Programm als Schrifttyp. Fehlt der Schrifttyp, wird er vom Programm auf D gesetzt.

Sollten Sie diese Funktion unbeabsichtigt angewählt haben, bringt die ESC-Taste Sie wieder in das Hauptmenü.

Als Listing 2 und 3 finden Sie bereits zwei fertige Zeichensätze.

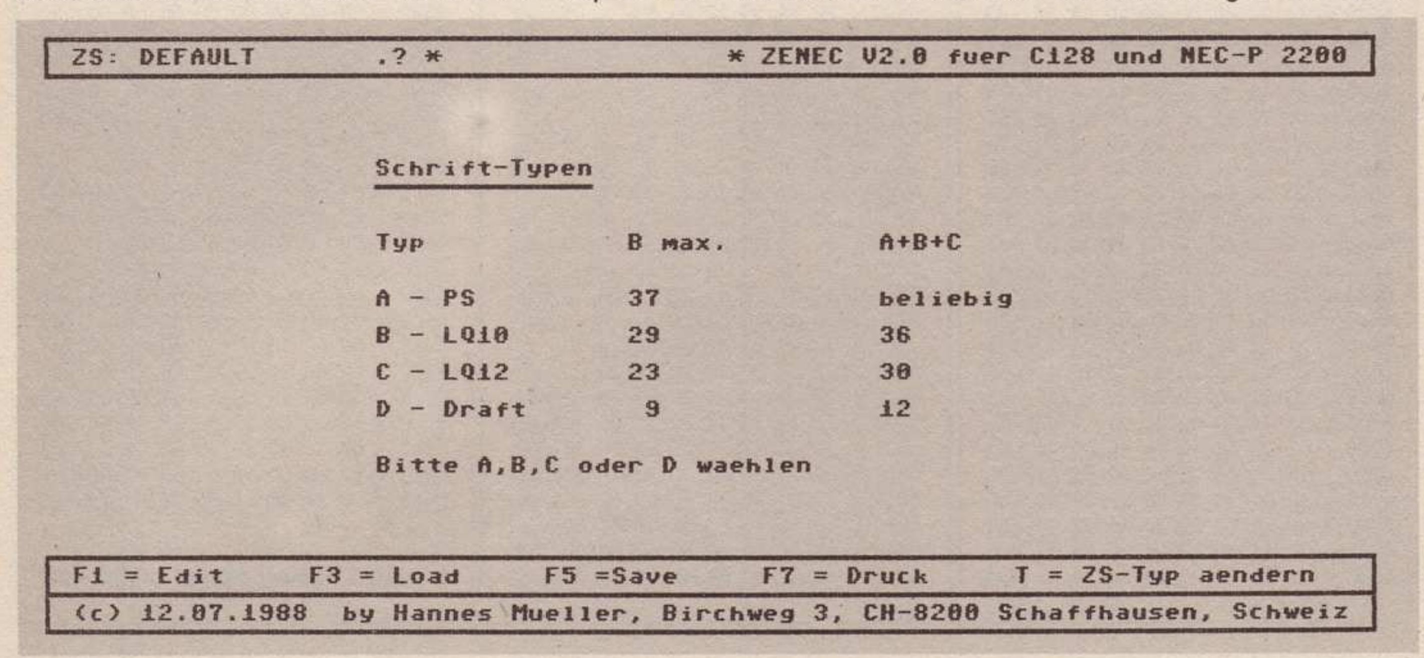


Bild 1. Im Eingangsmenü von »Zenec V2.0« bestimmen Sie den Schrifttyp, den Sie editieren

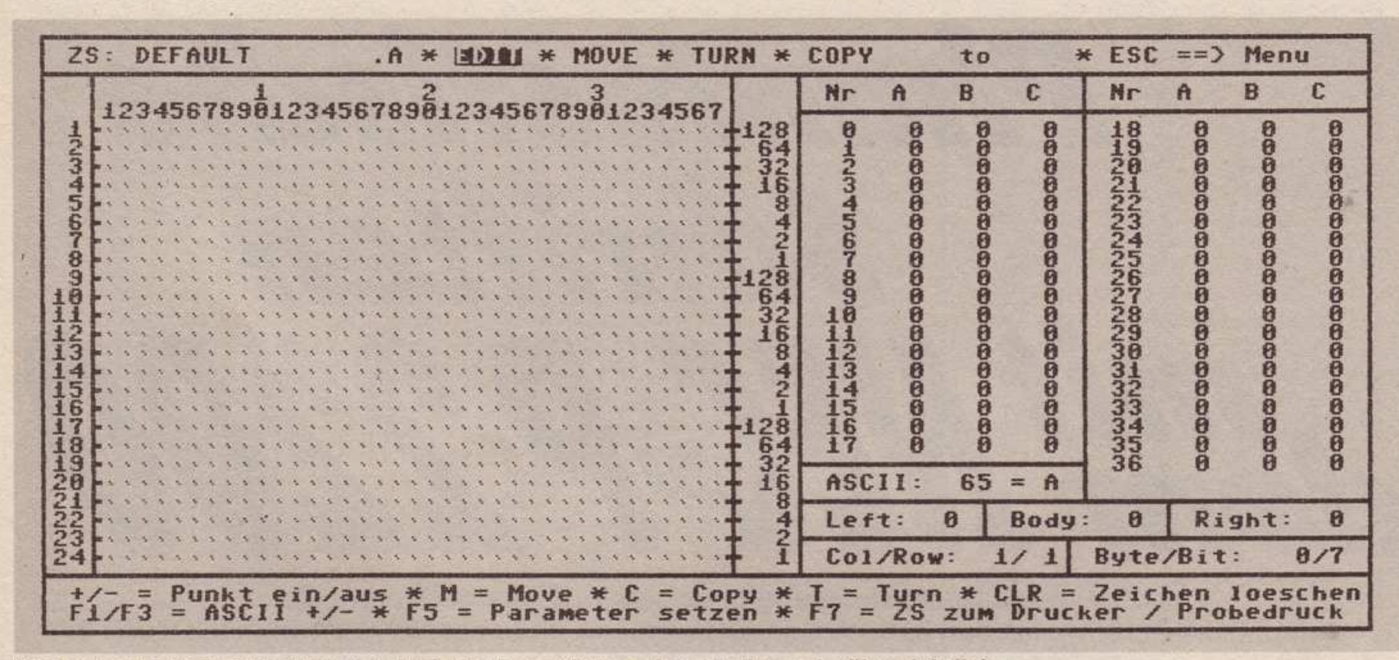


Bild 2. Der Editier-Bildschirm ist dank der Darstellung mit 33 Zeilen sehr übersichtlich

<F5> Save

Gespeichert wird ein editierter Zeichensatz über die Funktionstaste <F5>. Das Programm schlägt einen Filenamen (Default) vor, den Sie durch ein RETURN übernehmen können. Bei einer Neueingabe werden maximal 14 Zeichen angenommen, da das Programm automatisch einen Punkt und den im Eingangsmenü gewählten Schrifttyp-Buchstaben an den Dateinamen anhängt.

Achten Sie darauf, jeweils nach Änderungen eines Zeichensatzes einen neuen Dateinamen zu vergeben, da bei der Übernahme des alten Namens der alte Zeichensatz nicht überschrieben wird. Das Programm gibt dann eine Fehlermeldung aus. Lassen Sie sich von der flackernden LED des Laufwerks nicht irritieren, nach dem nächsten erfolgreichen Zugriff auf Diskette ist wieder alles in Ordnung.

<F7> Druck

Wenn Sie die Funktionstaste <F7> drücken, wird der aktuelle Zeichensatz zum Drucker übertragen. Dabei erfolgt gleichzeitig ein Probedruck des Zeichensatzes mit Angabe des ausgewählten Dateinamens.

Mit dieser Funktion wird auch ein Zeichensatz in den Drucker für die Anwendung unter einer Textverarbeitung geladen.

<T> Zeichensatz-Typ ändern

Haben Sie im Eingangsmenü bei der Auswahl des Schrifttyps eine falsche Eingabe gemacht, kehren Sie über die Taste <T> zu diesem Menü zurück. Wählen Sie dann den Schrifttyp, den Sie editieren wollen.

Ein komfortabler Editor

Sobald Sie vom Hauptmenü aus über die Funktionstaste <F1> den Editor angewählt haben, sehen Sie auf dem Bildschirm das Editierfeld. Wenn Sie vorher keinen fertigen Zeichensatz von Diskette geladen haben, ist das Feld leer (Bild 2). Ansonsten erscheint ein Zeichen aus einem vorher geladenen Zeichensatz (Bild 3).

Eine ausgezeichnete Programmierung des VDC erlaubt eine Bildschirmdarstellung von insgesamt 33 Zeilen bei 80 Spalten. Dieser Platz reicht aus, um alle Funktionen am Bildschirm sehr übersichtlich auszugeben.

Sehr hilfreich ist zum Beispiel die Anzeige der dezimalen Werte der einzelnen Bytes. Am unteren Rand erkennen Sie die Tastenbelegungen für die Bearbeitung der Zeichen. Der jeweils aktive Modus wird am oberen Bildschirmrand blinkend dargestellt.

Die Funktionen des Editors im Überblick:

Cursortasten

Mit den Cursortasten bewegen Sie sich wie gewohnt innerhalb des Editierfeldes.

<HOME>

Auch die HOME-Taste bewegt den Cursor wie gewohnt an die erste Position des Feldes.

<CLR>

Diese Taste löscht das gerade bearbeitete Zeichen.

<+/-> Punkt ein/aus

Mit diesen beiden Tasten setzen oder löschen Sie einen Punkt. Der Cursor bewegt sich dabei in der aktuellen Cursor-Richtung weiter. Dies ist zu Beginn sicher etwas gewöhnungsbedürftig, wenn Richtungsänderungen vorgenommen werden sollen. Hat man sich aber darauf eingestellt, wird das Editieren so recht einfach.

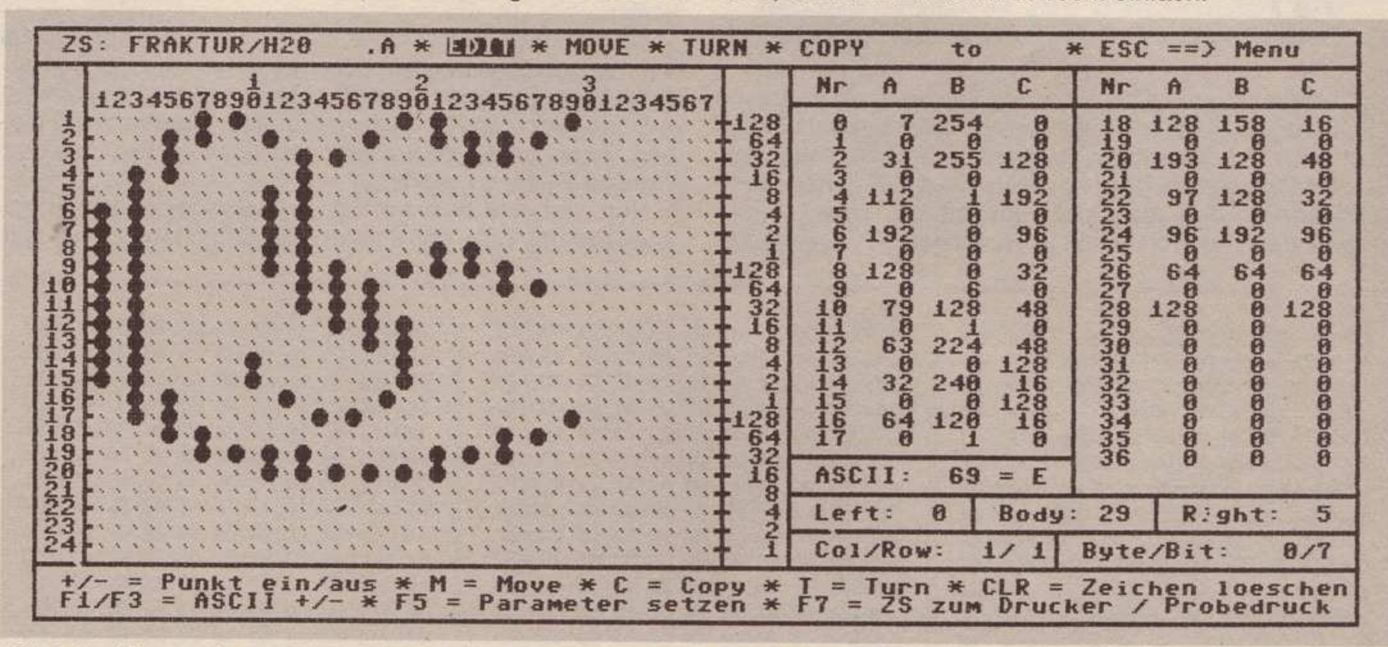


Bild 3. So könnte ein neues Zeichen aussehen. Editiert ist hier der Buchstabe <E>.

<F1> und <F3>

Diese beiden Funktionstasten werden dazu verwendet, um ein Zeichen vorwärts oder rückwärts zu springen. Im Feld ASCII sehen Sie dabei jeweils das ausgewählte Zeichen mit seinem dezimalen ASCII-Wert angezeigt.

<F5> Parameter setzen

Wählen Sie diesen Menüpunkt, springt der Cursor in die Anzeige der drei Parameter, die für linken Leerraum (Left), Zeichenkörper (Body) und rechten Leerraum (Right) stehen.

Mit den Cursortasten auf/ab werden die Werte verändert und mit RETURN bestätigt. Die Werte für die drei Parameter müssen den Schrifttypen entsprechen. Stehen die Werte auf Null, nutzt das Programm das entsprechende Zeichen aus dem Drucker-ROM.

Unsinnige Werte werden vom Programm automatisch für den Ausdruck zurechtgestutzt.

<F7> Zeichensatz zum Drucker /Probedruck

Der Zeichensatz wird zum Drucker geschickt, der sofort einen Probedruck beginnt. Dies ist die gleiche Funktion wie im Hauptmenü.

<M> Move

Über die Cursortasten wird ein bearbeitetes Zeichen in diesem Modus in alle vier Richtungen bewegt. Das Scrollen durch das ganze Editierfeld erlaubt vielfältige Variationen eines Zeichens. Mit der ESC-Taste wird dieser Modus wieder verlassen.

<C> Copy

Ein bearbeitetes Zeichen kann in diesem Modus an die Stelle eines anderen Zeichens kopiert werden. Wählen Sie dazu das Zielzeichen mit <CRSR-aufwärts/abwärts> aus. Mit RETURN wird der Kopiervorgang ausgeführt.

Der Modus kann mit <ESC> vorzeitig verlassen werden.

<T> Turn

Mit den Cursortasten wird ein bearbeitetes Zeichen um die horizontale oder vertikale Achse gespiegelt. Zum Editieren gelangen Sie mit <ESC> zurück.

<ESC>

Aus dem Editiermodus springen Sie mit dieser Taste ins Hauptmenü zurück.

Der parallele 24-Nadel-Drucker NEC-P 2200 erlaubt es, userdefinierte Zeichensätze zu verwenden. Der zugelassene Definitionsbereich für die neuen Zeichen geht dabei von ASCII 0 bis 127. Werden jedoch Werte unter dezimal 33 und oberhalb 126 verwendet, können eventuell bestimmte Control-Funktionen, die von den neuen Zeichen überschrieben werden, nicht mehr genutzt werden.

Die Fähigkeiten des Druckers nutzen

Die Zeichenmatrix umfaßt 37 Spalten und 24 Zeilen. Die Anordnung der Daten selbstdefinierter Zeichen ist zwingend vorgeschrieben:

Jeder der Spalten werden von oben nach unten 3 Bytes zugeordnet. Dabei entspricht ein gesetztes Bit einem gesetzten Punkt. Das Zeichen selbst wird in drei Teile gegliedert: A = linker Leerraum, B = Zeichenkörper und C = rechter Leerraum. Diese drei Parameter werden in Anzahl von Spalten angegeben. Für die verschiedenen Druckarten sind feste Größen vorgeschrieben, die Sie im Eingangsmenü als Vorgaben wiederfinden.

Tips für die Anwendung

Natürlich wollen Sie die editierten Zeichensätze ja auch innerhalb einer Textverarbeitung anwenden. Starten Sie dazu zunächst «Zenec v2.0» mit RUN von der Diskette und laden den entsprechenden Zeichensatz, den Sie verwenden wollen.

Nachdem Sie den Drucker eingeschaltet haben, wird der Zeichensatz zum Drucker geschickt. Nun wird der Computer ausgeschaltet und anschließend wieder eingeschaltet.

Der Drucker muß selbstverständlich eingeschaltet bleiben, da er sonst wieder die eingebauten Zeichen aus dem Drucker-ROM verwendet.

Nun wird eine Textverarbeitung geladen, innerhalb welcher Sie mit Hilfe der entsprechenden Steuercodes zwi-

```
Name : zenec v2.0          1c01 3bea
------------------------------------
1c01 : 0b 1c c4 07 9e 37 31 38  05
1c09 : 31 00 00 00 20 7d ff 0e  45
1c11 : 0b 99 00 ad 3e 03 c9 80  c3
1c19 : f0 2e 20 7d ff 93 c2 49  12
1c21 : 54 54 45 20 c1 d3 c3 c9  52
1c29 : c9 2f c4 c9 ce 2d d4 41  20
1c31 : 53 54 45 20 41 55 53 4c  a8
1c39 : bc 53 45 4e 21 00 ad 3e  ff
1c41 : 03 c9 80 f0 03 4c 3f 1c  2f
1c49 : ad c5 0a 09 80 8d c5 0a  1c
1c51 : 20 b3 77 20 30 1d a5 9d  ea
1c59 : 29 3f 85 9d 20 bd 1c a9  eb
1c61 : 0e 8d 00 ff 20 6e 1c 20  5c
1c69 : ed 1c 4c 34 38 a9 10 8d  2a
1c71 : 2f 0a a2 14 a9 10 20 cc  06
1c79 : cd a2 06 a9 21 20 cc cd  30
1c81 : a2 07 a9 25 20 cc cd a2  9b
1c89 : 0b a9 e9 20 cc cd a9 00  c9
1c91 : 85 fa a9 10 85 fb a9 85  ea
1c99 : 85 ff 20 aa 1c a9 00 85  95
1ca1 : fa a9 00 85 fb a9 20 85  b9
1ca9 : ff a9 50 85 fe a0 21 20  fb
1cb1 : 5e 1d 88 d0 01 60 20 6b  44
1cb9 : 1d 4c b3 1c a9 e5 85 fc  47
1cc1 : a9 1c 85 fd a9 0f 85 fe  c0
1cc9 : a9 01 85 ff a6 ff a0 01  43
1cd1 : a9 fc 20 65 ff e6 ff e6  b2
1cd9 : fc d0 02 e6 fd a5 ff c9  3b
1ce1 : 09 d0 e9 60 85 89 86 8a  ad
1ce9 : 87 8b 88 8c a9 e0 85 fa  97
1cf1 : a9 31 85 fb a9 20 85 fc  bf
1cf9 : a9 1d 85 fd 20 10 1d a9  9c
1d01 : f0 85 fa a9 31 85 fb a9  2a
1d09 : 28 85 fc a9 1d 85 fd 20  9e
1d11 : 44 1d a0 00 b1 fc 20 ca  25
1d19 : cd c8 c0 08 d0 f6 60 3c  3a
1d21 : 7e ff ff ff ff 7e 3c 00  83
1d29 : 00 00 10 08 00 00 00 a2  74
1d31 : 40 a0 00 84 fc 98 86 fd  fd
1d39 : 91 fc c8 d0 fb e8 e0 6b  f6
1d41 : 90 f4 60 a5 fa a2 13 20  69
```

```
1d49 : cc cd a5 fb ca 4c cc cd  c2
1d51 : a5 24 a2 13 20 cc cd a5  fe
1d59 : 25 ca 4c cc cd a2 18 20  23
1d61 : da cd 29 7f 20 cc cd 20  3c
1d69 : 44 1d a5 ff 20 ca cd a5  80
1d71 : fe 38 e9 01 ca 4c cc cd  04
1d79 : 20 e4 ff c9 00 f0 f9 60  75
1d81 : a5 24 a2 0f 20 cc cd a5  ae
1d89 : 25 ca 20 cc cd 4c 6f cd  4e
1d91 : 20 81 1d 20 79 1d 48 20  9f
1d99 : 9f cd 68 60 20 44 1d a0  1f
1da1 : 00 b1 fc f0 07 20 ca cd  0f
1da9 : c8 4c a2 1d 60 85 63 a9  f7
1db1 : 00 85 64 a9 00 85 65 85  8f
1db9 : 66 85 67 85 68 a5 64 c9  45
1dc1 : 03 d0 04 a5 63 c9 e8 b0  6c
1dc9 : 1b a5 64 c9 00 d0 04 a5  eb
1dd1 : 63 c9 64 b0 21 a5 63 c9  a8
1dd9 : 0a b0 2d a5 63 c9 01 b0  25
1de1 : 33 4c 1c 1e e6 65 38 a5  cb
1de9 : 63 e9 e8 85 63 a5 64 e9  f4
1df1 : 03 85 64 4c be 1d e6 66  97
1df9 : 38 a5 63 e9 64 85 63 a5  65
1e01 : 64 e9 00 85 64 4c ca 1d  19
1e09 : e6 67 38 a5 63 e9 0a 85  1e
1e11 : 63 4c d6 1d e6 68 c6 63  87
1e19 : 4c dc 1d a4 65 d0 19 a9  44
1e21 : 20 85 69 a4 66 d0 18 a9  93
1e29 : 20 85 6a a4 67 d0 17 a9  e8
1e31 : 20 85 6b a4 68 4c 4e 1e  e2
1e39 : b9 68 1e 85 69 a4 66 b9  27
1e41 : 68 1e 85 6a a4 67 b9 68  a4
1e49 : 1e 85 6b a4 68 b9 68 1e  cb
1e51 : 85 6c 60 a5 69 20 ca cd  37
1e59 : a5 6a 20 ca cd a5 6b 20  8d
1e61 : ca cd a5 6c 4c ca cd 30  bb
1e69 : 31 32 33 34 35 36 37 38  59
1e71 : 39 c2 01 c6 01 ca 01 12  b3
1e79 : 02 16 02 1a 02 62 02 66  52
1e81 : 02 6a 02 b2 02 b6 02 ba  e3
1e89 : 02 02 03 06 03 0a 03 52  3f
1e91 : 03 56 03 5a 03 a2 03 a6  6a
1e99 : 03 aa 03 f2 03 f6 03 fa  fa
```

```
1ea1 : 03 42 04 46 04 4a 04 92  57
1ea9 : 04 96 04 9a 04 e2 04 e6  82
1eb1 : 04 ea 04 32 05 36 05 3a  fc
1eb9 : 05 82 05 86 05 8a 05 d2  70
1ec1 : 05 d6 05 da 05 22 06 26  94
1ec9 : 06 2a 06 72 06 76 06 7a  d5
1ed1 : 06 c2 06 c6 06 ca 06 12  86
1ed9 : 07 16 07 1a 07 d3 01 d7  b3
1ee1 : 01 db 01 23 02 27 02 2b  2c
1ee9 : 02 73 02 77 02 7b 02 c3  a0
1ef1 : 02 c7 02 cb 02 13 03 17  c4
1ef9 : 03 1b 03 63 03 67 03 6b  05
1f01 : 03 b3 03 b7 03 bb 03 03  b6
1f09 : 04 07 04 0b 04 53 04 57  8d
1f11 : 04 5b 04 a3 04 a7 04 ab  1d
1f19 : 04 f3 04 f7 04 fb 04 43  ce
1f21 : 05 47 05 4b 05 93 05 97  a5
1f29 : 05 9b 05 e3 05 e7 05 eb  35
1f31 : 05 33 06 37 06 3b 06 83  92
1f39 : 06 87 06 8b 06 d3 06 d7  bd
1f41 : 06 db 06 23 07 27 07 2b  37
1f49 : 07 73 07 77 07 7b 07 54  cc
1f51 : 08 5f 08 6b 08 94 01 14  ca
1f59 : 04 94 06 95 01 15 04 95  d0
1f61 : 06 96 01 16 04 96 06 97  f2
1f69 : 01 17 04 97 06 98 01 18  43
1f71 : 04 98 06 99 01 19 04 99  92
1f79 : 06 9a 01 1a 04 9a 06 9b  b4
1f81 : 01 1b 04 9b 06 9c 01 1c  06
1f89 : 04 9c 06 9d 01 1d 04 9d  55
1f91 : 06 9e 01 1e 04 9e 06 9f  77
1f99 : 01 1f 04 9f 06 a0 01 20  c9
1fa1 : 04 a0 06 a1 01 21 04 a1  18
1fa9 : 06 a2 01 22 04 a2 06 a3  3a
1fb1 : 01 23 04 a3 06 a4 01 24  8b
1fb9 : 04 a4 06 a5 01 25 04 a5  da
1fc1 : 06 a6 01 26 04 a6 06 a7  fc
1fc9 : 01 27 04 a7 06 a8 01 28  4e
1fd1 : 04 a8 06 a9 01 29 04 a9  9d
1fd9 : 06 aa 01 2a 04 aa 06 ab  bf
1fe1 : 01 2b 04 ab 06 ac 01 2c  10
1fe9 : 04 ac 06 ad 01 2d 04 ad  60
1ff1 : 06 ae 01 2e 04 ae 06 af  82
```

schen dem Originalzeichensatz und Ihrem eigenen Zeichensatz umschalten können.

Achten Sie bei der Textverarbeitung darauf, daß Sie einen vorhanden Steuercode für einen Reset aus der Steuerzeichentabelle entfernen. Dies ist zum Beispiel bei »Mastertext 64« und bei »Mastertext 128« notwendig, da durch den Reset des Druckers der eigene Zeichensatz verloren geht.

Bei »Vizawrite« für den C128 ist die Einstellung des Druckers auf dem parallelen Anschluß vorzunehmen. Zusätzlich sollten Sie sowohl Linefeed als auch Formfeed einschalten.

»Protext 128« erlaubt die Verwendung des eigenen Zeichensatzes über den Druckertreiber »parallel-1«. Für einen guten Ausdruck ist auch hier ein zusätzliches Linefeed erforderlich.

Da »Zenec v2.0« nach dem Übertragen des Zeichensatzes sofort mit einem Probeausdruck beginnt, kann die erste Papierseite nicht mehr verwendet werden. Sie sollten diese Seite aber für einen ersten Testdruck benutzen. Dies ist sinnvoll, da manche Textverarbeitungsprogramme vor dem ersten Text einige Grafikzeichen ausdrucken. Sollten Sie das auch bei Ihrer Textverarbeitung feststellen, so braucht Sie das nicht zu irritieren. Nach dem Druck dieser Grafikzeichen stehen alle Zeichen Ihres Zeichensatzes für den Ausdruck zur Verfügung.

Das Programm ist vollständig in Assembler geschrieben. Es belegt dabei in der Bank 0 den Bereich von $1c01 bis $3be9 und ist mit einer Basic-Startzeile versehen.

Der Zeichenspeicher liegt ebenfalls in Bank 0, und zwar von $4000 bis $6ac1. Die beiden Basic-ROMs sind ausgeblendet. Der 40-Zeichen-Bildschirm dient als Hilfsspeicher.

Für die Darstellung der Menüs wird durch eine intensive Nutzung des Prozessors der benötigte Speicherplatz möglichst gering gehalten. Die Zeichenmatrix von maximal 37 Spalten und 24 Zeilen erfordert für jedes Zeichen 111 Byte zu 8 Punkten. Hinzu kommen die 3 Steuerbyte A, B und C. Für den Zeichensatz werden also 96 mal 114 Byte benötigt, die hintereinander im Speicher abgelegt sind und über eine Anfangsadreßtabelle mit einem Zeropage-Zeiger angesprochen werden.

Kurze Beschreibung des Programmaufbaus

Das Directory wird beim Ladevorgang erst in den Speicher ab $8000 eingelesen und dort aufbereitet. Beim Laden wird der Schrifttyp interpretiert. Die eigentlichen Load- und Save-Routinen sind standardisiert.

Da die Steuerbefehle für 24-Nadel-Drucker weitgehend standadisiert sind, dürfte »Zenec 2.0« auch mit anderen 24-Nadlern zusammenarbeiten. (Hannes Müller/ef)

```
1ff9 : 01 2f 04 af 06 b0 01 30  d3
2001 : 04 b0 06 b1 01 31 04 b1  22
2009 : 06 b2 01 32 04 b2 06 b3  44
2011 : 01 33 04 b3 06 b4 01 34  96
2019 : 04 b4 06 b5 01 35 04 b5  e5
2021 : 06 b6 01 36 04 b6 06 b7  07
2029 : 01 37 04 b7 06 b8 01 38  58
2031 : 04 b8 06 00 40 72 40 e4  75
2039 : 40 56 41 c8 41 3a 42 ac  56
2041 : 42 1e 43 90 43 02 44 74  b3
2049 : 44 e6 44 58 45 ca 45 3c  55
2051 : 46 ae 46 20 47 92 47 04  b2
2059 : 48 76 48 e8 48 5a 49 cc  22
2061 : 49 3e 4a b0 4a 22 4b 94  7e
2069 : 4b 06 4c 78 4c ea 4c 5c  df
2071 : 4d ce 4d 40 4e b2 4e 24  7d
2079 : 4f 96 4f 08 50 7a 50 ec  dc
2081 : 50 5e 51 d0 51 42 52 b4  49
2089 : 52 26 53 98 53 0a 54 7c  a6
2091 : 54 ee 54 60 55 d2 55 44  47
2099 : 56 b6 56 28 57 9a 57 0c  a5
20a1 : 58 7e 58 f0 58 62 59 d4  14
20a9 : 59 46 5a b8 5a 2a 5b 9c  71
20b1 : 5b 0e 5c 80 5c f2 5c 64  d2
20b9 : 5d d6 5d 48 5e ba 5e 2c  6f
20c1 : 5f 9e 5f 10 60 82 60 f4  cf
20c9 : 60 66 61 d8 61 4a 62 bc  3b
20d1 : 62 2e 63 a0 63 12 64 84  99
20d9 : 64 f6 64 68 65 da 65 4c  3a
20e1 : 66 be 66 30 67 a2 67 14  97
20e9 : 68 86 68 f8 68 6a 69 dc  07
20f1 : 69 4e 6a 6c 6d 6e 69 6a  6e
20f9 : 6b 66 67 68 63 64 65 60  2e
2101 : 61 62 5d 5e 5f 5a 5b 5c  a5
2109 : 57 58 59 54 55 56 51 52  5f
2111 : 53 4e 4f 50 4b 4c 4d 48  46
2119 : 49 4a 45 46 47 42 43 44  be
2121 : 3f 40 41 3c 3d 3e 39 3a  77
2129 : 3b 36 37 38 33 34 35 30  5e
2131 : 31 32 2d 2e 2f 2a 2b 2c  d6
2139 : 27 28 29 24 25 26 21 22  90
2141 : 23 1e 1f 20 1b 1c 1d 18  76
2149 : 19 1a 15 16 17 12 13 14  ee
2151 : 0f 10 11 0c 0d 0e 09 0a  a8
2159 : 0b 06 07 08 03 04 05 00  8f
2161 : 01 02 02 01 00 05 04 03  42
2169 : 08 07 06 0b 0a 09 0e 0d  13
2171 : 0c 11 10 0f 14 13 12 17  3c
2179 : 16 15 1a 19 18 1d 1c 1b  d5
2181 : 20 1f 1e 23 22 21 26 25  2b
2189 : 24 29 28 27 2c 2b 2a 2f  54
2191 : 2e 2d 32 31 30 35 34 33  ec
2199 : 38 37 36 3b 3a 39 3e 3d  43
21a1 : 3c 41 40 3f 44 43 42 47  6c
21a9 : 46 45 4a 49 48 4d 4c 4b  04
21b1 : 50 4f 4e 53 52 51 56 55  5b
21b9 : 54 59 58 57 5c 5b 5a 5f  84
21c1 : 5e 5d 62 61 60 65 64 63  1c
21c9 : 68 67 66 6b 6a 69 6e 6d  72
21d1 : 6c 00 80 40 c0 20 a0 60  b6
21d9 : e0 10 90 50 d0 30 b0 70  22
21e1 : f0 08 88 48 c8 28 a8 68  42
21e9 : e8 18 98 58 d8 38 b8 78  32
21f1 : f8 04 84 44 c4 24 a4 64  5e
21f9 : e4 14 94 54 d4 34 b4 74  42
2201 : f4 0c 8c 4c cc 2c ac 6c  62
2209 : ec 1c 9c 5c dc 3c bc 7c  52
2211 : fc 02 82 42 c2 22 a2 62  84
2219 : e2 12 92 52 d2 32 b2 72  62
2221 : f2 0a 8a 4a ca 2a aa 6a  82
2229 : ea 1a 9a 5a da 3a ba 7a  72
2231 : fa 06 86 46 c6 26 a6 66  9e
2239 : e6 16 96 56 d6 36 b6 76  82
2241 : f6 0e 8e 4e ce 2e ae 6e  a2
2249 : ee 1e 9e 5e de 3e be 7e  92
2251 : fe 01 81 41 c1 21 a1 61  c7
2259 : e1 11 91 51 d1 31 b1 71  a2
2261 : f1 09 89 49 c9 29 a9 69  c2
2269 : e9 19 99 59 d9 39 b9 79  b2
2271 : f9 05 85 45 c5 25 a5 65  de
2279 : e5 15 95 55 d5 35 b5 75  c2
2281 : f5 0d 8d 4d cd 2d ad 6d  e2
2289 : ed 1d 9d 5d dd 3d bd 7d  d2
2291 : fd 03 83 43 c3 23 a3 63  04
2299 : e3 13 93 53 d3 33 b3 73  e2
22a1 : f3 0b 8b 4b cb 2b ab 6b  02
22a9 : eb 1b 9b 5b db 3b bb 7b  f2
22b1 : fb 07 87 47 c7 27 a7 67  1e
22b9 : e7 17 97 57 d7 37 b7 77  02
22c1 : f7 0f 8f 4f cf 2f af 6f  22
22c9 : ef 1f 9f 5f df 3f bf 7f  12
22d1 : ff 5a 05 09 03 08 05 0e  01
22d9 : 13 01 14 1a 20 13 10 05  9a
22e1 : 09 03 08 05 12 0e 00 46  2d
22e9 : 09 0c 05 0e 01 0d 05 3a  fd
22f1 : 20 3c 00 44 45 46 41 55  ee
22f9 : 4c 54 20 20 20 20 20 20  3f
2301 : 20 2e 3f 00 3e 20 3a 20  16
2309 : 20 20 20 20 20 20 20 20  09
2311 : 20 20 20 20 20 20 20 20  11
2319 : 00 44 45 46 41 55 4c 54  ee
2321 : 20 20 20 20 20 20 20 2e  3d
2329 : 3f 00 52 09 03 08 14 09  f1
2331 : 07 20 3c 4e 3e 20 3f 00  03
2339 : a9 d2 85 fc a9 22 85 fd  0a
2341 : a9 95 85 fa a9 02 85 fb  2e
2349 : 20 9d 1d 20 8f 23 20 f7  06
2351 : 23 20 cb 23 20 d9 23 c9  cd
2359 : 4a f0 03 4c 7f 23 20 51  9a
2361 : 1d a9 4a 20 ca cd 20 c4  0f
2369 : 28 90 03 4c 8f 24 a5 90  f6
2371 : f0 03 4c 8f 24 20 6e 24  2d
2379 : 4c 54 38 4c 8f 23 c9 1b  f6
2381 : f0 03 4c 8c 23 4c 45 38  b2
2389 : 4c 8f 23 4c 4f 23 a0 00  80
2391 : b9 1a 23 99 f4 22 c8 c0  58
2399 : 10 d0 f5 20 aa 23 a9 1a  31
23a1 : 85 fc a9 23 85 fd 4c bd  68
23a9 : 23 a9 e8 85 fc a9 22 85  3c
23b1 : fd a9 35 85 fa a9 03 85  95
23b9 : fb 4c 9d 1d 20 a0 1d a9  b4
23c1 : 05 85 fc a9 23 85 fd 4c  ec
23c9 : a0 1d 20 aa 23 a9 f4 85  b4
```

Listing 1. Das Programm »Zenec V2.0« bitte mit dem MSE auf Seite 158 eingeben

```
23d1 : fc a9 22 85 fd 4c bd 23   5b
23d9 : a9 2b 85 fc a9 23 85 fd   de
23e1 : a9 d5 85 fa a9 03 85 fb   f6
23e9 : 20 9d 1d a9 de 85 24 a9   52
23f1 : 03 85 25 4c 91 1d a9 54   db
23f9 : 85 24 a9 03 85 25 a9 00   83
2401 : 85 27 20 91 1d c9 0d f0   8a
2409 : 03 4c 17 24 a5 27 f0 5c   8d
2411 : 4c 52 24 4c 38 24 c9 14   0d
2419 : f0 03 4c 2e 24 a5 24 d0   05
2421 : 02 c6 25 c6 24 c6 27 4c   56
2429 : 03 24 4c 38 24 c9 1b f0   37
2431 : 03 4c 38 24 4c 4b 2f a4   12
2439 : 27 99 f4 22 48 20 51 1d   b3
2441 : 68 20 ca cd e6 24 d0 02   fc
2449 : e6 25 e6 27 c8 c0 0e d0   cd
2451 : b1 a4 27 c0 0e f0 0a a9   1a
2459 : 20 99 f4 22 c8 c0 0e d0   34
2461 : f8 a9 2e 8d 02 23 ad 29   ad
2469 : 23 8d 03 23 60 a0 00 b9   f7
2471 : f4 22 99 1a 23 c8 c0 10   bc
2479 : d0 f5 60 a9 1a 85 fc a9   a6
2481 : 23 85 fd a9 56 85 fa a9   ec
2489 : 00 85 fb 4c 9d 1d a9 af   9d
2491 : 85 fc a9 24 85 fd a9 75   5d
2499 : 85 fa a9 04 85 fb 20 9d   7a
24a1 : 1d 20 79 1d c9 0d d0 f9   0d
24a9 : 20 e7 ff 4c 45 38 49 2f   e0
24b1 : 4f 2d 46 05 08 0c 05 12   e2
24b9 : 20 20 2d 2d 2d 3e 20 52   c4
24c1 : 45 54 55 52 4e 00 a9 20   9c
24c9 : 85 60 a9 80 85 61 20 8f   fc
24d1 : 1c 20 ab 28 90 03 4c 8f   5f
24d9 : 24 a9 00 85 71 a9 80 85   f4
24e1 : 72 20 f6 27 20 59 27 20   b0
24e9 : 03 28 20 f6 27 18 a9 20   01
24f1 : 65 71 85 71 a9 00 65 72   b3
24f9 : 85 72 20 7e 27 c9 20 d0   72
2501 : 0c 20 83 27 20 59 27 20   8d
2509 : 03 28 4c eb 24 20 83 27   50
2511 : 20 b2 27 20 a4 27 20 10   7c
2519 : 28 a9 17 85 fc a9 28 85   55
2521 : fd a9 64 85 fa a9 00 85   c5
2529 : fb 20 9d 1d a9 39 85 fc   b4
2531 : a9 28 85 fd a9 b4 85 fa   5c
2539 : a9 00 85 fb 20 9d 1d a9   7a
2541 : 5b 85 fc a9 28 85 fd a9   cd
2549 : 04 85 fa a9 01 85 fb 20   70
2551 : 9d 1d a9 10 85 70 a9 39   de
2559 : 85 fc a9 28 85 fd 18 a9   c8
2561 : 50 65 fa 85 fa a9 00 65   9b
2569 : fb 85 fb 20 9d 1d c6 70   e8
2571 : d0 ec a9 5b 85 fc a9 28   c4
2579 : 85 fd a9 54 85 fa a9 06   d5
2581 : 85 fb 20 9d 1d a9 39 85   cf
2589 : fc a9 28 85 fd a9 a4 85   df
2591 : fa a9 06 85 fb 20 9d 1d   03
2599 : a9 7d 85 fc a9 28 85 fd   f0
25a1 : a9 f4 85 fa a9 06 85 fb   5e
25a9 : 20 9d 1d a9 d3 85 fc a9   c5
25b1 : 27 85 fd a9 e4 85 fa a9   09
25b9 : 07 85 fb 20 9d 1d a9 00   ef
25c1 : 85 71 a9 80 85 72 a9 00   0c
25c9 : 85 fc a9 80 85 fd a9 b6   a3
25d1 : 85 fa a9 00 85 fb 20 9d   32
25d9 : 1d a9 c0 85 fc a9 27 85   70
25e1 : fd a9 a6 85 fa a9 06 85   2d
25e9 : fb 20 9d 1d a9 00 85 70   91
25f1 : a5 60 85 71 a5 61 85 72   b6
25f9 : a9 56 85 24 a9 01 85 25   b6
2601 : a0 00 b1 71 c9 2b f0 38   66
2609 : a5 70 c9 10 b0 60 a5 71   e2
2611 : 85 fc a5 72 85 fd a5 24   f3
2619 : 85 fa a5 25 85 fb 20 9d   1d
2621 : 1d 18 a9 20 65 71 85 71   94
2629 : a9 00 65 72 85 72 18 a9   1a
2631 : 50 65 24 85 24 a9 00 65   48
2639 : 25 85 25 e6 70 4c 01 26   01
2641 : a5 70 c9 10 b0 28 a9 3b   fc
2649 : 85 fc a9 28 85 fd a5 24   e3
2651 : 85 fa a5 25 85 fb 20 9d   55
2659 : 1d 18 a9 50 65 24 85 24   cd
2661 : a9 00 65 25 85 25 e6 70   06
2669 : a5 70 c9 10 90 d8 a9 56   de
2671 : 85 fa a9 11 85 fb a9 1d   19
2679 : 85 fe a9 cd 85 ff 20 5e   37
2681 : 1d a5 60 85 71 a5 61 85   0e
2689 : 72 18 a9 20 65 71 85 71   51
2691 : a9 00 65 72 85 72 20 79   41
2699 : 1d c9 11 f0 03 4c c5 26   f3
26a1 : a0 00 b1 71 c9 2b f0 03   9c
26a9 : 4c b2 26 4c 97 26 4c c2   c3
26b1 : 26 18 a9 20 65 60 85 60   82
26b9 : a9 00 65 61 85 61 4c ed   58
26c1 : 25 4c 28 27 c9 91 f0 03   ee
26c9 : 4c f4 26 a5 61 c9 80 d0   d6
26d1 : 04 a5 60 c9 20 f0 03 4c   27
26d9 : e1 26 4c 97 26 4c f1 26   ac
26e1 : 38 a5 60 e9 20 85 60 a5   3c
26e9 : 61 e9 00 85 61 4c ed 25   6a
26f1 : 4c 28 27 c9 1b f0 03 4c   32
26f9 : 04 27 20 8f 1c 4c 45 38   35
2701 : 4c 28 27 c9 0d f0 03 4c   61
2709 : 25 27 20 8f 1c 20 28 27   6e
2711 : 20 9f 28 90 03 4c 8f 24   36
2719 : 20 42 27 20 6e 24 4c 54   0a
2721 : 38 4c 28 27 4c 97 26 18   b9
2729 : a9 05 65 60 85 60 a9 00   bc
2731 : 65 61 85 61 a0 00 b1 60   66
2739 : 99 f4 22 c8 c0 10 d0 f6   ac
2741 : 60 ad 03 23 c9 41 90 0a   9a
2749 : c9 45 b0 06 8d 29 23 4c   e9
2751 : 6e 24 20 6e 24 4c 34 38   8d
2759 : a0 02 a2 00 b1 71 c8 c9   00
2761 : 22 d0 f9 b1 71 c9 22 f0   70
2769 : 08 9d 05 05 c8 e8 4c 64   f0
2771 : 27 c8 e8 b1 71 9d 05 05   8f
2779 : c9 00 d0 f5 60 a0 02 b1   ac
2781 : 71 60 a0 00 b1 71 85 63   ce
2789 : c8 b1 71 85 64 20 b4 1d   8b
2791 : a0 00 b9 69 00 99 00 05   a4
2799 : c8 c0 04 d0 f5 a9 20 99   3d
27a1 : 00 05 60 a0 00 b9 00 05   28
27a9 : 99 c0 27 c8 c0 04 d0 f5   e1
27b1 : 60 a0 00 b9 c0 27 99 04   4c
27b9 : 05 c8 c9 00 d0 f5 60 20   13
27c1 : 20 20 20 20 20 42 0c 0f   60
27c9 : 05 03 0b 05 20 06 12 05   38
27d1 : 09 00 57 01 08 0c 20 0d   4c
27d9 : 09 14 20 43 15 12 13 0f   a9
27e1 : 12 20 2a 20 4c 01 04 05   79
27e9 : 0e 20 0d 09 14 20 52 45   82
27f1 : 54 55 52 4e 00 a9 20 a0   5d
27f9 : 00 99 00 05 c8 c0 20 d0   1b
2801 : f8 60 a0 00 b9 00 05 91   24
2809 : 71 c8 c0 20 d0 f6 60 a9   ac
2811 : 2b a0 00 91 71 60 70 40   1b
2819 : 40 40 40 40 40 40 40 40   19
2821 : 40 40 40 40 40 40 40 40   21
2829 : 40 40 40 40 40 40 40 40   29
2831 : 40 40 40 40 40 40 6e 00   69
2839 : 5d 20 20 20 20 20 20 20   76
2841 : 20 20 20 20 20 20 20 20   41
2849 : 20 20 20 20 20 20 20 20   49
2851 : 20 20 20 20 20 20 20 20   51
2859 : 5d 00 6b 40 40 40 40 40   21
2861 : 40 40 40 40 40 40 40 40   61
2869 : 40 40 40 40 40 40 40 40   69
2871 : 40 40 40 40 40 40 40 40   71
2879 : 40 40 73 00 6d 40 40 40   11
2881 : 40 40 40 40 40 40 40 40   81
2889 : 40 40 40 40 40 40 40 40   89
2891 : 40 40 40 40 40 40 40 40   91
2899 : 40 40 40 40 7d 00 20 ef   4a
28a1 : 28 a9 00 a2 00 a0 40 4c   91
28a9 : d5 ff 20 d8 28 a9 01 a2   ba
28b1 : c3 a0 28 20 bd ff 20 e0   f1
28b9 : 28 a9 00 a2 00 a0 80 4c   aa
28c1 : d5 ff 24 20 ef 28 a9 00   8a
28c9 : 85 fc a9 40 85 fd a9 fc   28
28d1 : a2 c1 a0 6a 4c d8 ff a9   a8
28d9 : 00 a2 08 a8 4c ba ff a9   2f
28e1 : 00 aa 4c 68 ff a9 10 a2   29
28e9 : f4 a0 22 4c bd ff 20 d8   4d
28f1 : 28 20 e6 28 4c e0 28 a0   96
28f9 : dd 4c ff 28 a0 85 a9 77   cd
2901 : 85 fa a9 10 85 fb a2 04   3a
2909 : 86 fe 84 ff 4c 5e 1d a0   9d
2911 : dd 4c 17 29 a0 85 a9 70   bd
2919 : 85 fa a9 10 85 fb 4c 07   ff
2921 : 29 a0 dd 4c 2c 29 20 97   57
2929 : 1d a0 85 a9 69 85 fa a9   2f
2931 : 10 85 fb 4c 07 29 a0 dd   84
2939 : 4c 3e 29 a0 85 a9 7e 85   ad
2941 : fa a9 10 85 fb 20 07 29   f4
2949 : a5 ff 29 ef 85 ff a9 83   3c
2951 : 85 fa a9 10 85 fb a9 03   a5
2959 : 85 fe 20 5e 1d a9 8a 85   86
2961 : fa a9 10 85 fb 20 5e 1d   59
2969 : a9 83 85 fa a9 00 85 fb   3d
2971 : a9 20 85 ff a9 03 85 fe   52
2979 : 20 5e 1d a9 8a 85 fa a9   59
2981 : 00 85 fb 4c 5e 1d 20 cf   bb
2989 : 29 4c 6a 2a 00 00 00 00   b8
2991 : 00 00 00 00 a0 00 20 44   a5
2999 : 1d b9 8d 29 20 ca cd 18   db
29a1 : a9 50 65 fa 85 fa a9 00   02
29a9 : 65 fb 85 fb c8 c0 08 d0   41
29b1 : e5 60 85 52 a2 00 18 06   09
29b9 : 52 b0 03 4c c4 29 a9 1e   26
29c1 : 4c c6 29 a9 1f 9d 8d 29   57
29c9 : e8 e0 08 d0 e9 60 20 44   e8
29d1 : 2a 20 db 29 20 16 2a 4c   1b
29d9 : 25 2a a5 26 20 ae 1d a9   81
29e1 : b6 85 fa a9 07 85 fb 20   1a
29e9 : 44 1d 20 59 1e a9 bc 85   1c
29f1 : fa a9 07 85 fb 20 44 1d   3e
29f9 : a5 26 c9 40 f0 0d 90 0d   00
2a01 : c9 60 f0 0c 90 07 e9 60   62
2a09 : 4c ca cd a9 00 4c ca cd   8c
2a11 : a9 40 4c ca cd a0 6f 20   27
2a19 : 37 2a a0 70 20 37 2a a0   41
2a21 : 71 4c 37 2a a9 00 85 27   ca
2a29 : a4 27 20 39 2a e6 27 a9   5a
2a31 : 6f c5 27 d0 f3 60 84 27   09
2a39 : b1 6d 20 ae 1d 20 56 2a   ff
2a41 : 4c 59 1e 38 a5 26 e9 20   3c
2a49 : 0a aa bd 34 20 85 6d bd   fe
2a51 : 35 20 85 6e 60 38 a5 27   72
2a59 : 0a a8 b9 72 1e a2 13 20   f7
2a61 : cc cd b9 73 1e ca 4c cc   f4
2a69 : cd a9 00 85 5e a4 5e b1   a3
2a71 : 6d 20 b3 29 a5 5e 20 88   df
2a79 : 2a e6 5e a5 5e c9 6f d0   f6
2a81 : ec 60 20 b3 29 a5 27 0a   8c
2a89 : a8 b9 56 1f 85 fa b9 57   4d
2a91 : 1f 85 fb 4c 95 29 a9 01   46
2a99 : 85 29 a5 50 c9 25 f0 03   b6
2aa1 : 4c bf 2a a5 27 38 e9 6c   c1
2aa9 : 85 27 a9 01 85 50 38 a5   53
2ab1 : 24 e9 24 85 24 a5 25 e9   5b
2ab9 : 00 85 25 4c cd 2a a5 27   62
```

```
2ac1 : 69 03 85 27 e6 24 d0 02  c9
2ac9 : e6 25 e6 50 4c a7 38 a9  3c
2ad1 : 02 85 29 a5 50 c9 01 f0  ce
2ad9 : 03 4c f8 2a a5 27 18 69  4c
2ae1 : 6c 85 27 a9 25 85 50 18  ff
2ae9 : a9 24 65 24 85 24 a9 00  a2
2af1 : 65 25 85 25 4c 08 2b a5  ec
2af9 : 27 e9 03 85 27 c6 50 a5  bc
2b01 : 24 d0 02 c6 25 c6 24 4c  98
2b09 : a7 38 a9 03 85 29 a5 51  72
2b11 : c9 18 f0 03 4c 34 2b a9  e9
2b19 : 07 85 28 a9 01 85 51 c6  31
2b21 : 27 c6 27 38 a5 24 e9 30  00
2b29 : 85 24 a5 25 e9 07 85 25  06
2b31 : 4c 57 2b a5 28 c9 00 f0  5b
2b39 : 03 4c 46 2b a9 07 85 28  93
2b41 : e6 27 4c 48 2b c6 28 e6  2e
2b49 : 51 18 a9 50 65 24 85 24  f1
2b51 : a9 00 65 25 85 25 4c a7  fa
2b59 : 38 a9 04 85 29 a5 51 c9  b0
2b61 : 01 f0 03 4c 83 2b a9 18  8d
2b69 : 85 51 a9 00 85 28 e6 27  85
2b71 : e6 27 18 a9 30 65 24 85  f0
2b79 : 24 a9 07 65 25 85 25 4c  8c
2b81 : a6 2b a5 28 c9 07 f0 03  ca
2b89 : 4c 95 2b a9 00 85 28 c6  fa
2b91 : 27 4c 97 2b e6 28 c6 51  97
2b99 : 38 a5 24 e9 50 85 24 a5  f7
2ba1 : 25 e9 00 85 25 4c a7 38  2f
2ba9 : 20 bb 2b 1d 05 2c 4c c5  84
2bb1 : 2b 20 bb 2b 3d 0d 2c 4c  c6
2bb9 : c5 2b 20 97 1d a4 27 a6  f0
2bc1 : 28 b1 6d 60 91 6d 48 20  0f
2bc9 : 3b 2a 68 20 83 2a a5 29  aa
2bd1 : c9 01 f0 03 4c de 2b 4c  b8
2bd9 : 97 2a 4c 02 2c c9 02 f0  d4
2be1 : 03 4c eb 2b 4c d0 2a 4c  f7
2be9 : 02 2c c9 03 f0 03 4c f8  1e
2bf1 : 2b 4c 0b 2b 4c 02 2c c9  84
2bf9 : 04 f0 03 4c 02 2c 4c 5a  27
2c01 : 2b 4c 97 2a 01 02 04 08  be
2c09 : 10 20 40 80 fe fd fb f7  09
2c11 : ef df bf 7f 20 97 1d a9  56
2c19 : 55 85 fa a9 18 85 fb a9  16
2c21 : 6f 85 27 20 44 2c a9 60  2e
2c29 : 85 fa a9 18 85 fb e6 27  bb
2c31 : 20 44 2c a9 6c 85 fa a9  e6
2c39 : 18 85 fb e6 27 20 44 2c  cc
2c41 : 4c a4 38 20 44 1d a4 27  00
2c49 : b1 6d 85 53 a9 cd 20 ca  9c
2c51 : cd 20 cc cd 20 79 1d c9  f1
2c59 : 91 f0 03 4c 65 2c 4c 8f  b5
2c61 : 2c 4c 8f 2c c9 11 f0 03  0c
2c69 : 4c 72 2c 4c 9d 2c 4c 8f  0e
2c71 : 2c c9 0d f0 03 4c 7f 2c  cc
2c79 : 4c b1 2c 4c 8f 2c c9 1b  ea
2c81 : f0 03 4c 8c 2c 4c f7 38  0d
2c89 : 4c 8f 2c 4c 55 2c a5 53  25
2c91 : c9 25 90 03 4c 9a 2c e6  89
2c99 : 53 4c a9 2c a5 53 d0 03  40
2ca1 : 4c a6 2c c6 53 4c a9 2c  bb
2ca9 : a5 53 20 3b 2a 4c 55 2c  1a
2cb1 : a5 53 a4 27 91 6d 20 44  9b
2cb9 : 1d a9 85 20 ca cd 4c cc  f6
2cc1 : cd 20 97 1d a5 26 c9 7f  da
2cc9 : f0 03 4c d5 2c a9 20 85  a4
2cd1 : 26 4c d7 2c e6 26 4c 5a  1e
2cd9 : 38 20 97 1d a5 26 c9 20  9e
2ce1 : f0 03 4c ed 2c a9 7f 85  3d
2ce9 : 26 4c ef 2c c6 26 4c 5a  3a
2cf1 : 38 a9 94 85 24 a9 01 85  72
2cf9 : 25 a9 00 85 27 a9 07 85  8a
2d01 : 28 a9 01 85 29 85 51 85  fe
2d09 : 50 60 a9 09 85 fa a9 09  fe
```

```
2d11 : 85 fb 20 44 1d a5 27 20  00
2d19 : ae 1d 20 59 1e a9 0d 85  f8
2d21 : fa a9 09 85 fb 20 44 1d  ef
2d29 : a5 28 20 ae 1d 20 63 1e  5d
2d31 : a9 f8 85 fa a9 08 85 fb  00
2d39 : 20 44 1d a5 50 20 ae 1d  72
2d41 : 20 5e 1e a9 fb 85 fa a9  78
2d49 : 08 85 fb 20 44 1d a5 51  7d
2d51 : 20 ae 1d 4c 5e 1e 48 8d  ac
2d59 : 01 dd a9 3f 8d 02 dd a9  4f
2d61 : ff 8d 03 dd ad 00 dd 29  48
2d69 : fb 8d 00 dd ea 2c 01 dc  b4
2d71 : 10 12 ad 00 dd 09 04 8d  47
2d79 : 00 dd a9 10 2c 0d dd f0  59
2d81 : ec 68 18 60 68 38 60 20  be
2d89 : 9d 2d 20 d4 2e 20 ee 2d  59
2d91 : 20 e2 2e 20 f8 2e 20 ed  0f
2d99 : 2e 4c f8 2e ad 29 23 c9  35
2da1 : 41 f0 03 4c b2 2d a9 18  10
2da9 : 85 fa a9 2f 85 fb 4c de  23
2db1 : 2d c9 42 f0 03 4c c4 2d  71
2db9 : a9 1e 85 fa a9 2f 85 fb  54
2dc1 : 4c de 2d c9 43 f0 03 4c  61
2dc9 : d6 2d a9 24 85 fa a9 2f  5a
2dd1 : 85 fb 4c de 2d a9 2c 85  1f
2dd9 : fa a9 2f 85 fb a0 00 b1  4c
2de1 : fa c9 ff f0 07 20 57 2d  07
2de9 : c8 4c e0 2d 60 a5 26 85  8c
2df1 : 6f a9 20 85 26 a9 32 85  71
2df9 : fa a9 2f 85 fb 20 de 2d  db
2e01 : 20 44 2a 4c 4d 2e a5 26  81
2e09 : 8d 3b 2f 8d 3c 2f a9 38  06
2e11 : 85 fa a9 2f 85 fb 20 de  da
2e19 : 2d ad 6f 04 20 57 2d ad  46
2e21 : 70 04 20 57 2d 85 5e 06  0b
2e29 : 5e 65 5e 85 5e ad 71 04  a3
2e31 : 20 57 2d a0 00 b1 6d 20  e0
2e39 : 57 2d c8 c4 5e d0 f6 e6  08
2e41 : 26 a5 26 c9 80 d0 b9 a5  bd
2e49 : 6f 85 26 60 a0 6f b1 6d  38
2e51 : 99 00 04 c8 c0 72 d0 f6  d5
2e59 : 18 6d 70 04 18 6d 6f 04  77
2e61 : 8d 72 04 a9 00 cd 70 04  96
2e69 : d0 03 4c 40 2e ad 29 23  11
2e71 : c9 42 f0 03 4c 7d 2e a2  a7
2e79 : 00 4c a4 2e c9 43 f0 03  0f
2e81 : 4c 89 2e a2 01 4c a4 2e  d3
2e89 : c9 44 f0 03 4c 95 2e a2  80
2e91 : 02 4c a4 2e c9 41 f0 03  19
2e99 : 4c a2 2e 4c 07 2e 4c a4  a8
2ea1 : 2e a2 02 4c a7 2e bd cb  a5
2ea9 : 2e cd 70 04 f0 03 8d 70  99
2eb1 : 04 bd ce 2e cd 72 04 d0  2f
2eb9 : 03 4c 07 2e a9 00 8d 6f  1a
2ec1 : 04 bd d1 2e 8d 71 04 4c  eb
2ec9 : 07 2e 1d 17 09 24 1e 0c  54
2ed1 : 07 07 03 a2 00 bd 1a 23  0e
2ed9 : f0 2e 20 57 2d e8 4c d6  cc
2ee1 : 2e a9 3e 85 fa a9 2f 85  e9
2ee9 : fb 4c de 2d a9 42 85 fa  20
2ef1 : a9 2f 85 fb 4c de 2d a2  c8
2ef9 : 20 8a c9 7f f0 07 20 57  37
2f01 : 2d e8 4c fa 2e 20 57 2d  b0
2f09 : a9 0d 20 57 2d a9 0a 20  b4
2f11 : 57 2d a9 0a 4c 57 2d 1c  17
2f19 : 40 1b 70 01 ff 1c 40 1b  3b
2f21 : 78 01 ff 1c 40 1b 78 01  5e
2f29 : 1b 4d ff 1c 40 1b 78 00  2d
2f31 : ff 1b 3a 00 00 00 ff 1b  83
2f39 : 26 00 00 00 ff 1b 25 01  cf
2f41 : ff 0d 0a 1c 56 01 1c 45  35
2f49 : 01 ff 20 8f 1c a9 00 85  5e
2f51 : fa a9 00 85 fb 20 ba 30  dd
2f59 : a9 f5 85 fc a9 30 85 fd  2c
```

```
2f61 : a9 50 85 fa a9 00 85 fb  9c
2f69 : 20 9d 1d a9 a0 85 fa a9  4a
2f71 : 00 85 fb 20 c7 30 a9 c0  5d
2f79 : 85 fa a9 08 85 fb 20 ba  15
2f81 : 30 a9 46 85 fc a9 31 85  b5
2f89 : fd a9 10 85 fa a9 09 85  3c
2f91 : fb 20 9d 1d a9 60 85 fa  51
2f99 : a9 09 85 fb 20 d4 30 a9  64
2fa1 : 97 85 fc a9 31 85 fd a9  fa
2fa9 : b0 85 fa a9 09 85 fb 20  fd
2fb1 : 9d 1d a9 00 85 fa a9 0a  32
2fb9 : 85 fb 20 c7 30 4c 7c 24  dc
2fc1 : 20 4b 2f a9 00 85 fe a5  fb
2fc9 : fe 0a a8 b9 ec 37 85 fa  c2
2fd1 : b9 ed 37 85 fb b9 10 38  3e
2fd9 : 85 fc b9 11 38 85 fd 20  55
2fe1 : 9d 1d e6 fe a9 12 c5 fe  e6
2fe9 : d0 dd 60 20 8f 1c a9 00  44
2ff1 : 85 fa a9 00 85 fb 20 ba  8c
2ff9 : 30 a9 e8 85 fc a9 31 85  d5
3001 : fd a9 50 85 fa a9 00 85  a0
3009 : fb 20 9d 1d a9 89 85 fc  16
3011 : a9 32 85 fd a9 f0 85 fa  22
3019 : a9 00 85 fb 20 9d 1d a9  5a
3021 : 00 85 fe a5 fe 0a a8 b9  ae
3029 : 2e 33 85 fa b9 2f 33 85  9e
3031 : fb b9 9a 33 85 fc b9 9b  74
3039 : 33 85 fd 20 9d 1d e6 fe  0f
3041 : a9 36 c5 fe d0 dd a9 59  ac
3049 : 85 fa a9 07 85 fb a9 6e  52
3051 : 85 fc a9 35 85 fd 20 44  b7
3059 : 1d a0 00 b1 fc c9 00 f0  fd
3061 : 03 4c 85 30 18 a9 50 65  cd
3069 : fa 85 fa a9 00 65 fb 85  40
3071 : fb 18 a9 2c 65 fc 85 fc  b6
3079 : a9 00 65 fd 85 fd 4c 57  63
3081 : 30 4c 99 30 c9 ff f0 03  aa
3089 : 4c 92 30 4c 99 30 4c 99  33
3091 : 30 20 ca cd c8 4c 5c 30  fe
3099 : a9 72 85 fc a9 36 85 fd  db
30a1 : a9 10 85 fa a9 09 85 fb  04
30a9 : 20 9d 1d a9 00 85 fa a9  80
30b1 : 0a 85 fb 20 c7 30 4c 7c  a9
30b9 : 24 20 44 1d a9 70 20 e1  04
30c1 : 30 a9 6e 4c ca cd 20 44  0f
30c9 : 1d a9 6d 20 e1 30 a9 7d  5b
30d1 : 4c ca cd 20 44 1d a9 6b  a4
30d9 : 20 e1 30 a9 73 4c ca cd  8b
30e1 : 20 ca cd e6 fa d0 02 e6  c2
30e9 : fb a9 40 85 ff a9 4e 85  0b
30f1 : fe 4c 5e 1d 5d 20 5a 53  37
30f9 : 3a 20 20 20 20 20 20 20  13
3101 : 20 20 20 20 20 20 20 20  01
3109 : 20 20 20 2a 20 20 20 20  4b
3111 : 20 20 20 20 20 20 20 20  11
3119 : 20 20 20 20 20 2a 20 5a  de
3121 : 45 4e 45 43 20 56 32 2e  21
3129 : 30 20 06 15 05 12 20 43  76
3131 : 31 32 38 20 15 0e 04 20  a0
3139 : 4e 45 43 2d 50 20 32 32  d4
3141 : 30 30 20 5d 00 5d 20 46  35
3149 : 31 20 3d 20 45 04 09 14  9e
3151 : 20 20 20 20 20 46 33 20  cf
3159 : 3d 20 4c 0f 01 04 20 20  8c
3161 : 20 20 20 46 35 20 3d 53  52
3169 : 01 16 05 20 20 20 20 20  7f
3171 : 46 37 20 3d 20 44 12 15  99
3179 : 03 0b 20 20 20 20 20 54  3a
3181 : 20 3d 20 5a 53 2d 54 19  b5
3189 : 10 20 01 05 0e 04 05 12  c4
```

Listing 1. »Zeichengenerator NEC-P 2000« (Fortsetzung)

```
3191 : 0e 20 20 20 5d 00 5d 20   47
3199 : 28 03 29 20 31 32 2e 30   4f
31a1 : 37 2e 31 39 38 38 20 20   69
31a9 : 02 19 20 48 01 0e 0e 05   0c
31b1 : 13 20 4d 15 05 0c 0c 05   b5
31b9 : 12 2c 20 42 09 12 03 08   6f
31c1 : 17 05 07 20 33 2c 20 43   bc
31c9 : 48 2d 38 32 30 30 20 53   a8
31d1 : 03 08 01 06 06 08 01 15   a8
31d9 : 13 05 0e 2c 20 53 03 08   31
31e1 : 17 05 09 1a 20 5d 00 5d   a8
31e9 : 20 5a 53 3a 20 20 20 20   16
31f1 : 20 20 20 20 20 20 20 20   f1
31f9 : 20 20 20 20 20 20 2a 20   21
3201 : 45 44 49 54 20 2a 20 4d   b4
3209 : 4f 56 45 20 2a 20 54 55   78
3211 : 52 4e 20 2a 20 43 4f 50   d2
3219 : 59 20 20 20 20 20 14 0f   00
3221 : 20 20 20 20 20 2a 20 45   bc
3229 : 53 43 20 3d 3d 3e 20 4d   ae
3231 : 05 0e 15 20 20 20 5d 6b   d6
3239 : 40 40 72 40 40 40 40 40   c6
3241 : 40 40 40 40 40 40 40 40   41
3249 : 40 40 40 40 40 40 40 40   49
3251 : 40 40 40 40 40 40 40 40   51
3259 : 40 40 40 40 40 40 40 40   59
3261 : 72 40 40 40 72 40 40 40   b6
3269 : 40 40 40 40 40 40 40 40   69
3271 : 40 40 40 40 40 72 40 40   03
3279 : 40 40 40 40 40 40 40 40   79
3281 : 40 40 40 40 40 40 73 00   cd
3289 : 5d 20 20 5d 20 20 20 20   6e
3291 : 20 20 20 20 20 31 20 20   1a
3299 : 20 20 20 20 20 20 20 32   bd
32a1 : 20 20 20 20 20 20 20 20   a1
32a9 : 20 33 20 20 20 20 20 20   33
32b1 : 20 5d 20 20 20 5d 20 4e   96
32b9 : 12 20 20 41 20 20 20 42   14
32c1 : 20 20 20 43 20 20 5d 20   1b
32c9 : 4e 12 20 20 41 20 20 20   02
32d1 : 42 20 20 20 43 20 20 5d   a0
32d9 : 5d 20 20 5d 31 32 33 34   d4
32e1 : 35 36 37 38 39 30 31 32   44
32e9 : 33 34 35 36 37 38 39 30   c5
32f1 : 31 32 33 34 35 36 37 38   e1
32f9 : 39 30 31 32 33 34 35 36   f3
3301 : 37 5d 20 20 20 6b 40 40   d2
3309 : 40 40 40 40 40 40 40 40   09
3311 : 40 40 40 40 40 40 5b 40   7d
3319 : 40 40 40 40 40 40 40 40   19
3321 : 40 40 40 40 40 40 40 73   87
3329 : 5d 20 31 6b 00 b9 01 ce   c0
3331 : 01 df 01 09 02 1e 02 2f   fb
3339 : 02 59 02 6e 02 7f 02 a9   ae
3341 : 02 be 02 cf 02 f9 02 0e   31
3349 : 03 1f 03 49 03 5e 03 6f   d4
3351 : 03 99 03 ae 03 bf 03 e9   c6
3359 : 03 fe 03 0f 04 39 04 4e   35
3361 : 04 5f 04 89 04 9e 04 af   ec
3369 : 04 d9 04 ee 04 ff 04 29   db
3371 : 05 3e 05 4f 05 79 05 8e   0e
3379 : 05 9f 05 c9 05 de 05 ef   04
3381 : 05 19 06 2e 06 3f 06 69   a0
3389 : 06 7e 06 8f 06 b9 06 ce   26
3391 : 06 df 06 09 07 1e 07 2f   05
3399 : 07 06 34 0f 34 14 34 1a   7b
33a1 : 34 23 34 28 34 2e 34 37   6d
33a9 : 34 3c 34 42 34 4b 34 50   60
33b1 : 34 56 34 5f 34 64 34 6a   15
33b9 : 34 73 34 78 34 7e 34 87   da
33c1 : 34 8c 34 92 34 9b 34 a0   cd
33c9 : 34 a6 34 af 34 b4 34 ba   83
33d1 : 34 c3 34 c8 34 ce 34 d7   47
33d9 : 34 dc 34 e2 34 eb 34 f0   3a
33e1 : 34 f6 34 ff 34 04 35 0a   ea
33e9 : 35 13 35 18 35 1e 35 27   60
33f1 : 35 2c 35 32 35 3b 35 40   52
33f9 : 35 46 35 4f 35 54 35 5a   08
3401 : 35 63 35 68 35 5b 31 32   9a
3409 : 38 5d 20 20 30 00 5d 20   b5
3411 : 31 38 00 5d 5d 20 32 6b   81
3419 : 00 5b 20 36 34 5d 20 20   85
3421 : 31 00 5d 20 31 39 00 5d   45
3429 : 5d 20 33 6b 00 5b 20 33   92
3431 : 32 5d 20 20 32 00 5d 20   f7
3439 : 32 30 00 5d 5d 20 34 6b   ae
3441 : 00 5b 20 31 36 5d 20 20   2c
3449 : 33 00 5d 20 32 31 00 5d   3f
3451 : 5d 20 35 6b 00 5b 20 20   15
3459 : 38 5d 20 20 34 00 5d 20   45
3461 : 32 32 00 5d 5d 20 36 6b   df
3469 : 00 5b 20 20 34 5d 20 20   12
3471 : 35 00 5d 20 32 33 00 5d   79
3479 : 5d 20 37 6b 00 5b 20 20   bd
3481 : 32 5d 20 20 36 00 5d 20   87
3489 : 32 34 00 5d 5d 20 38 6b   10
3491 : 00 5b 20 20 31 5d 20 20   0a
3499 : 37 00 5d 20 32 35 00 5d   b3
34a1 : 5d 20 39 6b 00 5b 31 32   ce
34a9 : 38 5d 20 20 38 00 5d 20   d5
34b1 : 32 36 00 5d 5d 31 30 6b   a1
34b9 : 00 5b 20 36 34 5d 20 20   25
34c1 : 39 00 5d 20 32 37 00 5d   ed
34c9 : 5d 31 31 6b 00 5b 20 33   3a
34d1 : 32 5d 20 31 30 00 5d 20   99
34d9 : 32 38 00 5d 5d 31 32 6b   d2
34e1 : 00 5b 20 31 36 5d 20 31   ee
34e9 : 31 00 5d 20 32 39 00 5d   1d
34f1 : 5d 31 33 6b 00 5b 20 20   bd
34f9 : 38 5d 20 31 32 00 5d 20   e7
3501 : 33 30 00 5d 5d 31 34 6b   ff
3509 : 00 5b 20 20 34 5d 20 31   d4
3511 : 33 00 5d 20 33 31 00 5d   17
3519 : 5d 31 35 6b 00 5b 20 20   65
3521 : 32 5d 20 31 34 00 5d 20   29
3529 : 33 32 00 5d 5d 31 36 6b   30
3531 : 00 5b 20 20 31 5d 20 31   cc
3539 : 35 00 5d 20 33 33 00 5d   51
3541 : 5d 31 37 6b 00 5b 31 32   76
3549 : 38 5d 20 31 36 00 5d 20   77
3551 : 33 34 00 5d 5d 31 38 6b   61
3559 : 00 5b 20 36 34 5d 20 31   e7
3561 : 37 00 5d 20 33 35 00 5d   8b
3569 : 5d 31 39 6b 00 5b 20 33   dc
3571 : 32 6b 40 40 40 40 40 40   f9
3579 : 40 40 40 40 40 40 40 40   79
3581 : 40 40 73 20 33 36 20 2d   82
3589 : 2d 2d 20 2d 2d 2d 20 2d   12
3591 : 2d 2d 20 5d 5d 32 30 6b   08
3599 : 00 5b 20 31 36 5d 20 41   c6
35a1 : 53 43 49 49 3a 20 27 27   a1
35a9 : 27 20 3d 20 27 20 5d 20   5d
35b1 : 20 20 20 20 20 20 20 20   b1
35b9 : 20 20 20 20 20 20 20 5d   34
35c1 : 5d 32 31 6b 00 5b 20 20   8d
35c9 : 38 6b 40 40 40 40 40 40   57
35d1 : 40 40 40 40 72 40 40 40   f4
35d9 : 40 40 71 40 40 40 40 72   8a
35e1 : 40 40 40 40 40 40 40 40   e1
35e9 : 40 40 40 73 5d 32 32 6b   cf
35f1 : 00 5b 20 20 34 5d 20 4c   f2
35f9 : 05 06 14 3a 20 27 27 20   66
3601 : 5d 20 42 0f 04 19 3a 20   13
3609 : 27 27 20 5d 20 52 09 07   3e
3611 : 08 14 3a 20 27 27 20 5d   9d
3619 : 5d 32 33 6b 00 5b 20 20   65
3621 : 32 6b 40 40 40 40 40 40   a9
3629 : 40 40 40 40 71 40 40 40   3c
3631 : 40 72 40 40 40 40 40 71   ac
3639 : 40 40 40 40 40 40 40 40   39
3641 : 40 40 40 73 5d 32 34 6b   2f
3649 : 00 5b 20 20 31 5d 20 43   08
3651 : 0f 0c 2f 52 0f 17 3a 20   4f
3659 : 27 27 2f 27 20 5d 20 42   b7
3661 : 19 14 05 2f 42 09 14 3a   dd
3669 : 20 27 27 27 2f 27 20 5d   33
3671 : ff 6b 40 40 71 40 40 40   d9
3679 : 40 40 40 40 40 40 40 40   79
3681 : 40 40 40 40 40 40 40 40   81
3689 : 40 40 40 40 40 40 40 40   89
3691 : 40 40 40 40 40 40 40 40   91
3699 : 40 40 71 40 40 40 71 40   aa
36a1 : 40 40 40 40 40 40 40 40   a1
36a9 : 40 40 40 40 40 40 71 40   6e
36b1 : 40 40 40 40 40 40 40 40   b1
36b9 : 40 40 40 40 40 40 40 40   b9
36c1 : 73 5d 20 2b 2f 2d 20 3d   a8
36c9 : 20 50 15 0e 0b 14 20 05   f4
36d1 : 09 0e 2f 01 15 13 20 2a   8c
36d9 : 20 4d 20 3d 20 4d 0f 16   24
36e1 : 05 20 2a 20 43 20 3d 20   ef
36e9 : 43 0f 10 19 20 2a 20 54   58
36f1 : 20 3d 20 54 15 12 0e 20   9d
36f9 : 2a 20 43 4c 52 20 3d 20   e9
3701 : 5a 05 09 03 08 05 0e 20   a2
3709 : 0c 0f 05 13 03 08 05 0e   e1
3711 : 5d 5d 20 46 31 2f 46 33   fa
3719 : 20 3d 20 41 53 43 49 49   0f
3721 : 20 2b 2f 2d 20 2a 20 46   a9
3729 : 35 20 3d 20 50 01 12 01   19
3731 : 0d 05 14 05 12 20 13 05   df
3739 : 14 1a 05 0e 20 2a 20 46   be
3741 : 37 20 3d 20 5a 53 20 1a   d1
3749 : 15 0d 20 44 12 15 03 0b   61
3751 : 05 12 20 2f 20 50 12 0f   38
3759 : 02 05 04 12 15 03 0b 20   f7
3761 : 5d 00 53 03 08 12 09 06   35
3769 : 14 2d 54 19 10 05 0e 00   ae
3771 : 40 40 40 40 40 40 40 40   71
3779 : 40 40 40 40 40 00 54 19   79
3781 : 10 00 42 20 0d 01 18 2e   bc
3789 : 00 41 2b 42 2b 43 00 41   8c
3791 : 20 2d 20 50 53 00 33 37   ca
3799 : 00 02 05 0c 09 05 02 09   30
37a1 : 07 00 42 20 2d 20 4c 51   e5
37a9 : 31 30 00 32 39 00 33 36   06
37b1 : 00 43 20 2d 20 4c 51 31   0d
37b9 : 32 00 32 33 00 33 30 00   39
37c1 : 44 20 2d 20 44 12 01 06   4a
37c9 : 14 00 20 39 00 31 32 00   5f
37d1 : 42 09 14 14 05 20 41 2c   ce
37d9 : 42 2c 43 20 0f 04 05 12   50
37e1 : 20 44 20 17 01 05 08 0c   7f
37e9 : 05 0e 00 44 02 94 02 84   54
37f1 : 03 93 03 a2 03 74 04 83   be
37f9 : 04 92 04 14 05 23 05 32   ac
3801 : 05 b4 05 c3 05 d2 05 54   be
3809 : 06 63 06 72 06 44 07 63   f6
3811 : 37 71 37 7f 37 83 37 8a   40
3819 : 37 90 37 97 37 9a 37 a3   c6
3821 : 37 ac 37 af 37 b2 37 bb   cf
3829 : 37 be 37 c1 37 cb 37 ce   12
3831 : 37 d1 37 20 c1 2f 20 79   2c
3839 : 1d c9 41 90 f9 c9 45 b0   02
3841 : f5 8d 29 23 a9 41 85 26   b3
3849 : 85 6f 20 44 2a 20 4b 2f   46
3851 : 4c 60 38 20 ec 2f 20 22   ec
3859 : 29 20 87 29 4c a4 38 20   a4
3861 : 79 1d c9 85 f0 03 4c 70   c5
```

```
3869 : 38 4c 54 38 4c a1 38 c9  2a
3871 : 86 f0 03 4c 7d 38 4c c7  14
3879 : 24 4c a1 38 c9 87 f0 03  d5
3881 : 4c 8a 38 4c 39 23 4c a1  cb
3889 : 38 c9 88 f0 03 4c 97 38  47
3891 : 20 88 2d 4c a1 38 c9 54  76
3899 : f0 03 4c a1 38 4c 34 38  79
38a1 : 4c 60 38 20 f2 2c 20 81  44
38a9 : 1d 20 0b 2d 20 79 1d c9  15
38b1 : 1d f0 03 4c c0 38 20 97  0e
38b9 : 1d 4c 97 2a 4c 9b 39 c9  42
38c1 : 9d f0 03 4c d0 38 20 97  9f
38c9 : 1d 4c d0 2a 4c 9b 39 c9  a0
38d1 : 11 f0 03 4c e0 38 20 97  24
38d9 : 1d 4c 0b 2b 4c 9b 39 c9  5f
38e1 : 91 f0 03 4c f0 38 20 97  b5
38e9 : 1d 4c 5a 2b 4c 9b 39 c9  42
38f1 : 1b f0 03 4c 00 39 20 97  48
38f9 : 1d 4c 4e 38 4c 9b 39 c9  f1
3901 : 85 f0 03 4c 0d 39 4c c2  9a
3909 : 2c 4c 9b 39 c9 86 f0 03  04
3911 : 4c 1a 39 4c da 2c 4c 9b  ba
3919 : 39 c9 87 f0 03 4c 27 39  d8
3921 : 4c 15 2c 4c 9b 39 c9 88  48
3929 : f0 03 4c 3a 39 20 97 1d  22
3931 : 20 88 2d 4c 54 38 4c 9b  da
3939 : 39 c9 13 f0 03 4c 4a 39  68
3941 : 20 97 1d 4c a4 38 4c 9b  72
3949 : 39 c9 93 f0 03 4c 57 39  cc
3951 : 4c d9 3b 4c 9b 39 c9 2b  e3
3959 : f0 03 4c 64 39 4c a9 2b  5d
3961 : 4c 9b 39 c9 2d f0 03 4c  01
3969 : 71 39 4c b2 2b 4c 9b 39  d6
3971 : c9 4d f0 03 4c 7e 39 4c  b4
3979 : 9b 39 4c 9b 39 c9 54 f0  4c
3981 : 03 4c 8b 39 4c ee 39 4c  6e
3989 : 9b 39 c9 43 f0 03 4c 98  25
3991 : 39 4c 41 3a 4c 9b 39 4c  a7
3999 : aa 38 20 27 29 20 10 29  73
```

```
39a1 : 20 79 1d c9 91 f0 03 4c  44
39a9 : b1 39 4c 8e 3a 4c ee 39  10
39b1 : c9 11 f0 03 4c be 39 4c  d8
39b9 : af 3a 4c ee 39 c9 9d f0  b0
39c1 : 03 4c cb 39 4c f1 3a 4c  da
39c9 : ee 39 c9 1d f0 03 4c d8  74
39d1 : 39 4c d5 3a 4c ee 39 c9  a2
39d9 : 1b f0 03 4c eb 39 20 15  ea
39e1 : 29 20 22 29 4c a4 38 4c  2b
39e9 : ee 39 4c a1 39 20 27 29  3f
39f1 : 20 f8 28 20 79 1d c9 91  66
39f9 : f0 03 4c 04 3a 4c 24 3b  0c
3a01 : 4c 41 3a c9 11 f0 03 4c  f3
3a09 : 11 3a 4c 24 3b 4c 41 3a  5f
3a11 : c9 9d f0 03 4c 1e 3a 4c  7d
3a19 : 0f 3b 4c 41 3a c9 1d f0  49
3a21 : 03 4c 2b 3a 4c 0f 3b 4c  1f
3a29 : 41 3a c9 1b f0 03 4c 3e  32
3a31 : 3a 20 fd 28 20 22 29 4c  50
3a39 : a4 38 4c 41 3a 4c f4 39  81
3a41 : 20 27 29 20 37 29 a5 26  e3
3a49 : 85 6f 20 79 3b 20 89 3b  0e
3a51 : 20 79 1d c9 91 f0 03 4c  f4
3a59 : 61 3a 4c 9c 3b 4c 8e 3a  43
3a61 : c9 11 f0 03 4c 6e 3a 4c  09
3a69 : b1 3b 4c 8e 3a c9 1b f0  dd
3a71 : 03 4c 7e 3a 20 3c 29 20  4a
3a79 : 5a 38 4c 8e 3a c9 0d f0  dc
3a81 : 03 4c 8b 3a 4c c9 3b 4c  6d
3a89 : 8e 3a 4c 51 3a 20 3d 3b  82
3a91 : a2 00 18 1e 02 04 3e 01  38
3a99 : 04 3e 00 04 b0 03 4c a5  dd
3aa1 : 3a fe 02 04 e8 e8 e8 e0  97
3aa9 : 6f d0 e7 4c 67 3b 20 3d  4f
3ab1 : 3b a2 00 18 5e 00 04 7e  33
3ab9 : 01 04 7e 02 04 b0 03 4c  07
3ac1 : cb 3a bd 00 04 09 80 9d  df
3ac9 : 00 04 e8 e8 e8 e0 6f d0  17
3ad1 : e2 4c 67 3b 20 44 2a a0  29
```

```
3ad9 : 00 b1 6d 99 03 04 c8 c0  35
3ae1 : 6c d0 f6 b1 6d 99 94 03  a5
3ae9 : c8 c0 6f d0 f6 4c 67 3b  ed
3af1 : 20 44 2a a0 03 b1 6d 99  79
3af9 : fd 03 c8 c0 6f d0 f6 a0  5d
3b01 : 00 b1 6d 99 6c 04 c8 c0  f4
3b09 : 03 d0 f6 4c 67 3b 20 44  15
3b11 : 2a a0 00 be f4 20 b1 6d  55
3b19 : 9d 00 04 c8 c0 6f d0 f3  83
3b21 : 4c 70 3b 20 44 2a a0 00  90
3b29 : b1 6d aa bd d2 21 be 63  eb
3b31 : 21 9d 00 04 c8 c0 6f d0  93
3b39 : ef 4c 70 3b 20 44 2a a0  e0
3b41 : 00 b1 6d 99 00 04 c8 c0  6d
3b49 : 72 d0 f6 60 a0 00 b9 00  de
3b51 : 04 91 6d c8 c0 6f d0 f6  4b
3b59 : 60 a0 00 b9 00 04 91 6d  82
3b61 : c8 c0 72 d0 f6 60 20 4d  cd
3b69 : 3b 20 87 29 4c a1 39 20  b2
3b71 : 5a 3b 20 87 29 4c f4 39  9d
3b79 : a9 83 85 fa a9 00 85 fb  4d
3b81 : 20 44 1d a5 26 4c 96 3b  55
3b89 : a9 8a 85 fa a9 00 85 fb  e1
3b91 : 20 44 1d a5 6f 20 ae 1d  bc
3b99 : 4c 59 1e a5 6f c9 7f f0  f3
3ba1 : 03 4c ac 3b a9 20 85 6f  ed
3ba9 : 4c ae 3b e6 6f 4c c3 3b  d7
3bb1 : a5 6f c9 20 f0 03 4c c1  60
3bb9 : 3b a9 7f 85 6f 4c c3 3b  38
3bc1 : c6 6f 20 89 3b 4c 51 3a  48
3bc9 : 20 3d 3b a5 6f 85 26 20  07
3bd1 : 44 2a 20 5a 3b 4c 75 3a  de
3bd9 : 20 97 1d a0 00 98 91 6d  06
3be1 : c8 c0 72 d0 f9 4c 5a 38  9c
3be9 : 56 c9 04 d0 0f a0 14 20  c6
```

Listing 1. »Zeichengenerator NEC-P 2000« (Schluß)

```
Name : fraktur/h20   .a  4000 6ac1
-----------------------------------
4000 : 00 00 00 00 00 00 00 00  01
4008 : 00 00 00 00 00 00 00 00  09
4010 : 00 00 00 00 00 00 00 00  11
4018 : 00 00 00 00 00 00 00 00  19
4020 : 00 00 00 00 00 00 00 00  21
4028 : 00 00 00 00 00 00 00 00  29
4030 : 00 00 00 00 00 00 00 00  31
4038 : 00 00 00 00 00 00 00 00  39
4040 : 00 00 00 00 00 00 00 00  41
4048 : 00 00 00 00 00 00 00 00  49
4050 : 00 00 00 00 00 00 00 00  51
4058 : 00 00 00 00 00 00 00 00  59
4060 : 00 00 00 00 00 00 00 00  61
4068 : 00 00 00 00 00 00 00 00  69
4070 : 15 05 1f 80 20 60 70 00  a7
4078 : 9f 8e 70 60 70 00 1f 80  0b
4080 : 20 00 00 00 00 00 00 00  a1
4088 : 00 00 00 00 00 00 00 00  89
4090 : 00 00 00 00 00 00 00 00  91
4098 : 00 00 00 00 00 00 00 00  99
40a0 : 00 00 00 00 00 00 00 00  a1
40a8 : 00 00 00 00 00 00 00 00  a9
40b0 : 00 00 00 00 00 00 00 00  b1
40b8 : 00 00 00 00 00 00 00 00  b9
40c0 : 00 00 00 00 00 00 00 00  c1
40c8 : 00 00 00 00 00 00 00 00  c9
40d0 : 00 00 00 00 00 00 00 00  d1
40d8 : 00 00 00 00 00 00 00 00  d9
40e0 : 00 0a 05 05 40 00 00 00  cb
```

```
40e8 : 00 00 c8 00 00 10 00 00  9b
40f0 : 60 00 00 00 00 00 00 00  51
40f8 : 00 00 00 00 00 00 00 00  f9
4100 : 00 00 40 00 00 00 00 00  11
4108 : c8 00 00 10 00 00 60 00  54
4110 : 00 00 00 00 00 00 00 00  11
4118 : 00 00 00 00 00 00 00 00  19
4120 : 00 00 00 00 00 00 00 00  21
4128 : 00 00 00 00 00 00 00 00  29
4130 : 00 00 00 00 00 00 00 00  31
4138 : 00 00 00 00 00 00 00 00  39
4140 : 00 00 00 00 00 00 00 00  41
4148 : 00 00 00 00 00 00 00 00  49
4150 : 00 00 00 00 0f 05 7f fe  66
4158 : 00 00 00 00 e3 ff ff 00  97
4160 : 00 00 c0 82 04 00 00 00  21
4168 : 41 82 02 00 00 00 61 04  f9
4170 : 02 00 00 00 61 84 06 00  c5
4178 : 00 00 40 fb fc 00 00 00  d8
4180 : 80 71 f8 00 00 00 00 00  f7
4188 : 00 00 00 00 00 00 00 00  89
4190 : 00 00 00 00 00 00 00 00  91
4198 : 00 00 00 00 00 00 00 00  99
41a0 : 00 00 00 00 00 00 00 00  a1
41a8 : 00 00 00 00 00 00 00 00  a9
41b0 : 00 00 00 00 00 00 00 00  b1
41b8 : 00 00 00 00 00 00 00 00  b9
41c0 : 00 00 00 00 00 00 0f 05  07
41c8 : 00 00 00 00 00 00 00 00  c9
41d0 : 00 00 00 00 00 00 00 00  d1
41d8 : 00 00 00 00 00 00 00 00  d9
```

```
41e0 : 00 00 00 00 00 00 00 00  e1
41e8 : 00 00 00 00 00 00 00 00  e9
41f0 : 00 00 00 00 00 00 00 00  f1
41f8 : 00 00 00 00 00 00 00 00  f9
4200 : 00 00 00 00 00 00 00 00  01
4208 : 00 00 00 00 00 00 00 00  09
4210 : 00 00 00 00 00 00 00 00  11
4218 : 00 00 00 00 00 00 00 00  19
4220 : 00 00 00 00 00 00 00 00  21
4228 : 00 00 00 00 00 00 00 00  29
4230 : 00 00 00 00 00 00 00 00  31
4238 : 00 00 01 00 00 00 00 00  79
4240 : ff ff ff 00 00 00 80 ff  41
4248 : 00 00 00 00 c0 00 00 00  55
4250 : 00 00 60 00 00 00 00 00  69
4258 : 60 00 00 00 00 00 40 00  ba
4260 : 00 00 00 00 80 00 00 00  69
4268 : 00 00 00 00 00 00 00 00  69
4270 : 00 00 00 00 00 00 00 00  71
4278 : 00 00 00 00 00 00 00 00  79
4280 : 00 00 00 00 00 00 00 00  81
4288 : 00 00 00 00 00 00 00 00  89
4290 : 00 00 00 00 00 00 00 00  91
```

Listing 2. Den Zeichensatz »FRAKTUR/H20.A« bitte mit dem MSE (Seite 158) eingeben. Vor dem Punkt bitte drei Leerzeichen eingeben.

```
4298 : 00 00 00 00 00 00 00 00  99
42a0 : 00 00 00 00 00 00 00 00  a1
42a8 : 00 00 0f 00 00 00 00 00  6c
42b0 : 00 00 00 00 00 00 00 00  b1
42b8 : 00 00 00 00 00 00 00 00  b9
42c0 : 00 00 00 00 00 00 00 00  c1
42c8 : 00 00 00 00 00 00 00 00  c9
42d0 : 00 00 00 00 00 00 00 00  d1
42d8 : 00 00 00 00 00 00 00 00  d9
42e0 : 00 00 00 00 00 00 00 00  e1
42e8 : 00 00 00 00 00 00 00 00  e9
42f0 : 00 00 00 00 00 00 00 00  f1
42f8 : 00 00 00 00 00 00 00 00  f9
4300 : 00 00 00 00 00 00 00 00  01
4308 : 00 00 00 00 00 00 00 00  09
4310 : 00 00 00 00 00 00 00 00  11
4318 : 00 00 00 00 00 00 00 00  19
4320 : 00 00 00 00 00 00 00 00  21
4328 : 00 00 00 00 00 00 00 00  29
4330 : 00 00 00 00 00 00 00 00  31
4338 : 00 00 00 00 00 00 00 00  39
4340 : 00 00 00 00 00 00 00 00  41
4348 : 00 00 00 00 00 00 00 00  49
4350 : 00 00 00 00 00 00 00 00  51
4358 : 00 00 00 00 00 00 00 00  59
4360 : 00 00 00 00 00 00 00 00  61
4368 : 00 00 00 00 00 00 00 00  69
4370 : 00 00 00 00 00 00 00 00  71
4378 : 00 00 00 00 00 00 00 00  79
4380 : 00 00 00 00 00 00 00 00  81
4388 : 00 00 00 00 00 00 00 00  89
4390 : 00 00 00 00 00 00 00 00  91
4398 : 00 00 00 00 00 00 00 00  99
43a0 : 00 00 00 00 00 00 00 00  a1
43a8 : 00 00 00 00 00 00 00 00  a9
43b0 : 00 00 00 00 00 00 00 00  b1
43b8 : 00 00 00 00 00 00 00 00  b9
43c0 : 00 00 00 00 00 00 00 00  c1
43c8 : 00 00 00 00 00 00 00 00  c9
43d0 : 00 00 00 00 00 00 00 00  d1
43d8 : 00 00 00 00 00 00 00 00  d9
43e0 : 00 00 00 00 00 00 00 00  e1
43e8 : 00 00 00 00 00 00 00 00  e9
43f0 : 00 00 00 00 00 00 00 00  f1
43f8 : 00 00 00 00 00 00 00 00  f9
4400 : 00 00 00 00 00 00 00 00  01
4408 : 00 00 00 00 00 00 00 00  09
4410 : 00 00 00 00 00 00 00 00  11
4418 : 00 00 00 00 00 00 00 00  19
4420 : 00 00 00 00 00 00 00 00  21
4428 : 00 00 00 00 00 00 00 00  29
4430 : 00 00 00 00 00 00 00 00  31
4438 : 00 00 00 00 00 00 00 00  39
4440 : 00 00 00 00 00 00 00 00  41
4448 : 00 00 00 00 00 00 00 00  49
4450 : 00 00 00 00 00 00 00 00  51
4458 : 00 00 00 00 00 00 00 00  59
4460 : 00 00 00 00 00 00 00 00  61
4468 : 00 00 00 00 00 00 00 00  69
4470 : 00 00 00 00 01 00 00 00  81
4478 : 00 00 ff ff ff 00 00 00  78
4480 : 80 ff 00 00 00 00 c0 00  03
4488 : 00 00 00 00 60 00 00 00  8f
4490 : 00 00 60 00 00 00 00 00  a9
4498 : 40 00 00 00 00 00 80 00  db
44a0 : 00 00 00 00 00 00 00 00  a1
44a8 : 00 00 00 00 00 00 00 00  a9
44b0 : 00 00 00 00 00 00 00 00  b1
44b8 : 00 00 00 00 00 00 00 00  b9
44c0 : 00 00 00 00 00 00 00 00  c1
44c8 : 00 00 00 00 00 00 00 00  c9
44d0 : 00 00 00 00 00 00 00 00  d1
44d8 : 00 00 00 00 00 00 00 00  d9
44e0 : 00 00 00 00 0f 05 00 08  0a
44e8 : 00 00 00 00 00 08 00 00  29
44f0 : 00 00 00 08 00 00 00 00  f2
44f8 : 00 08 00 00 00 00 01 ff  01
4500 : 80 00 00 00 00 ff c0 00  83
4508 : 00 00 00 08 00 00 00 00  0a
4510 : 00 08 00 00 00 00 00 08  25
4518 : 00 00 00 00 00 08 00 00  59
4520 : 00 00 00 00 00 00 00 00  21
4528 : 00 00 00 00 00 00 00 00  29
4530 : 00 00 00 00 00 00 00 00  31
4538 : 00 00 00 00 00 00 00 00  39
4540 : 00 00 00 00 00 00 00 00  41
4548 : 00 00 00 00 00 00 00 00  49
4550 : 00 00 00 00 00 00 13 05  a7
4558 : 00 00 20 00 00 00 00 00  61
4560 : 72 00 00 04 00 00 38 00  34
4568 : 00 00 00 00 00 00 00 00  69
4570 : 00 00 00 00 00 00 00 00  71
4578 : 00 00 00 00 00 00 00 00  79
4580 : 00 00 00 00 00 00 00 00  81
4588 : 00 00 00 00 00 00 00 00  89
4590 : 00 00 00 00 00 00 00 00  91
4598 : 00 00 00 00 00 00 00 00  99
45a0 : 00 00 00 00 00 00 00 00  a1
45a8 : 00 00 00 00 00 00 00 00  a9
45b0 : 00 00 00 00 00 00 00 00  b1
45b8 : 00 00 00 00 00 00 00 00  b9
45c0 : 00 00 00 00 00 00 00 05  cb
45c8 : 05 0a 00 08 00 00 00 00  d4
45d0 : 00 08 00 00 00 00 00 04  dd
45d8 : 00 00 00 00 00 04 00 00  f9
45e0 : 00 00 00 04 00 00 00 00  61
45e8 : 00 0c 00 00 00 00 00 0c  07
45f0 : 00 00 00 00 00 08 00 00  31
45f8 : 00 00 00 08 00 00 00 00  fa
4600 : 00 04 00 00 00 00 00 00  03
4608 : 00 00 00 00 00 00 00 00  09
4610 : 00 00 00 00 00 00 00 00  11
4618 : 00 00 00 00 00 00 00 00  19
4620 : 00 00 00 00 00 00 00 00  21
4628 : 00 00 00 00 00 00 00 00  29
4630 : 00 00 00 00 00 00 00 00  31
4638 : 00 00 13 05 00 00 20 00  1e
4640 : 00 00 00 00 70 00 00 00  48
4648 : 00 00 20 00 00 00 00 00  51
4650 : 00 00 00 00 00 00 00 00  51
4658 : 00 00 00 00 00 00 00 00  59
4660 : 00 00 00 00 00 00 00 00  61
4668 : 00 00 00 00 00 00 00 00  69
4670 : 00 00 00 00 00 00 00 00  71
4678 : 00 00 00 00 00 00 00 00  79
4680 : 00 00 00 00 00 00 00 00  81
4688 : 00 00 00 00 00 00 00 00  89
4690 : 00 00 00 00 00 00 00 00  91
4698 : 00 00 00 00 00 00 00 00  99
46a0 : 00 00 00 00 00 00 00 00  a1
46a8 : 00 00 00 05 05 0a 00 00  ea
46b0 : 00 00 00 00 00 00 00 00  b1
46b8 : 00 00 00 00 00 00 00 00  b9
46c0 : 00 00 00 00 00 00 00 00  c1
46c8 : 00 00 00 00 00 00 00 00  c9
46d0 : 00 00 00 00 00 00 00 00  d1
46d8 : 00 00 00 00 00 00 00 00  d9
46e0 : 00 00 00 00 00 00 00 00  e1
46e8 : 00 00 00 00 00 00 00 00  e9
46f0 : 00 00 00 00 00 00 00 00  f1
46f8 : 00 00 00 00 00 00 00 00  f9
4700 : 00 00 00 00 00 00 00 00  01
4708 : 00 00 00 00 00 00 00 00  09
4710 : 00 00 00 00 00 00 00 00  11
4718 : 00 00 00 00 00 00 00 00  19
4720 : 03 fc 00 00 00 00 0f ff  de
4728 : 00 00 00 00 1c 03 80 00  04
4730 : 00 00 30 00 c0 00 00 00  49
4738 : 60 00 60 00 00 00 40 00  b2
4740 : 20 00 00 00 80 00 10 00  a9
4748 : 00 00 80 00 10 00 00 00  6a
4750 : 80 00 10 00 00 00 80 00  d7
4758 : 10 00 00 00 40 00 20 00  ed
4760 : 00 00 60 00 60 00 00 00  7f
4768 : 30 00 c0 00 00 00 1c 03  3f
4770 : 80 00 00 00 0f ff 00 00  e1
4778 : 00 00 03 fc 00 00 00 00  d9
4780 : 00 00 00 00 00 00 00 00  81
4788 : 00 00 00 00 00 00 00 00  89
4790 : 1f 05 04 00 00 00 00 00  33
4798 : 04 00 10 00 00 00 0c 00  d1
47a0 : 10 00 00 00 1c 00 10 00  b3
47a8 : 00 00 3f ff f0 00 00 00  87
47b0 : ff ff f0 00 00 00 00 00  eb
47b8 : 10 00 00 00 00 00 10 00  09
47c0 : 00 00 00 00 10 00 00 00  c2
47c8 : 00 00 00 00 00 00 00 00  c9
47d0 : 00 00 00 00 00 00 00 00  d1
47d8 : 00 00 00 00 00 00 00 00  d9
47e0 : 00 00 00 00 00 00 00 00  e1
47e8 : 00 00 00 00 00 00 00 00  e9
47f0 : 00 00 00 00 00 00 00 00  f1
47f8 : 00 00 00 00 00 00 00 00  f9
4800 : 00 00 11 05 1e 00 30 00  88
4808 : 00 00 3f 00 70 00 00 00  df
4810 : 63 00 f0 00 00 00 46 01  cb
4818 : b0 00 00 00 80 03 30 00  a9
4820 : 00 00 80 06 30 00 00 00  04
4828 : 80 0c 30 00 00 00 80 18  ed
4830 : 30 00 00 00 80 30 30 00  ab
4838 : 00 00 80 60 30 00 00 00  68
4840 : 40 c0 30 00 00 00 61 80  73
4848 : 30 00 00 00 3f 01 a0 00  f7
4850 : 00 00 1e 00 c0 00 00 00  e4
4858 : 00 00 00 00 00 00 00 00  59
4860 : 00 00 00 00 00 00 00 00  61
4868 : 00 00 00 00 00 00 00 00  69
4870 : 00 00 00 00 1b 05 00 03  50
4878 : 80 00 00 00 0c 07 c0 00  f4
4880 : 00 00 3e 06 60 00 00 00  d7
4888 : 66 00 20 00 00 00 40 00  f8
4890 : 30 00 00 00 80 00 10 00  09
4898 : 00 00 80 40 10 00 00 00  c2
48a0 : 80 40 10 00 00 00 80 40  c7
48a8 : 30 00 00 00 40 e0 20 00  64
48b0 : 00 00 61 b0 60 00 00 00  25
48b8 : 3f 1f c0 00 00 00 1e 0f  4e
48c0 : 80 00 00 00 00 00 00 00  41
48c8 : 00 00 00 00 00 00 00 00  c9
48d0 : 00 00 00 00 00 00 00 00  d1
48d8 : 00 00 00 00 00 00 00 00  d9
48e0 : 00 00 00 00 00 00 19 05  4f
48e8 : 00 0c 00 00 00 00 00 1c  27
48f0 : 00 00 00 00 00 34 00 00  92
48f8 : 00 00 00 64 00 00 00 00  85
4900 : 00 c4 00 00 00 00 01 84  70
4908 : 00 00 00 00 03 04 00 00  59
4910 : 00 00 06 04 00 00 00 00  13
4918 : 0c 04 10 00 00 00 18 04  93
4920 : 10 00 00 00 30 04 10 00  94
4928 : 00 00 7f ff f0 00 00 00  17
4930 : ff ff f0 00 00 00 00 04  73
4938 : 10 00 00 00 00 04 10 00  a9
4940 : 00 00 00 04 00 00 00 00  c1
4948 : 00 00 00 00 00 00 00 00  49
4950 : 00 00 00 00 00 00 00 00  51
4958 : 1f 05 ff 81 80 00 00 00  32
4960 : ff 83 c0 00 00 00 81 01  59
4968 : 60 00 00 00 83 00 20 00  81
4970 : 00 00 82 00 30 00 00 00  14
4978 : 82 00 10 00 00 00 82 00  09
4980 : 10 00 00 00 82 00 10 00  f9
```

```
4988 : 00 00 83 00 30 00 00 00 6c
4990 : 81 00 20 00 00 00 81 80 21
4998 : 60 00 00 00 80 c0 c0 00 0a
49a0 : 00 00 80 7f 80 00 00 00 b8
49a8 : 80 1e 00 00 00 00 00 00 38
49b0 : 00 00 00 00 00 00 00 00 b1
49b8 : 00 00 00 00 00 00 00 00 b9
49c0 : 00 00 00 00 00 00 00 00 c1
49c8 : 00 00 1b 05 0f ff 00 00 21
49d0 : 00 00 3f ff c0 00 00 00 ac
49d8 : 70 70 e0 00 00 00 40 40 3a
49e0 : 20 00 00 00 c0 c0 30 00 d3
49e8 : 00 00 80 80 10 00 00 00 1a
49f0 : 80 80 10 00 00 00 80 80 b8
49f8 : 10 00 00 00 80 80 10 00 55
4a00 : 00 00 80 c0 30 00 00 00 3c
4a08 : c0 40 20 00 00 00 4c 70 03
4a10 : e0 00 00 00 7c 3f c0 00 b5
4a18 : 00 00 38 0f 00 00 00 00 08
4a20 : 00 00 00 00 00 00 00 00 21
4a28 : 00 00 00 00 00 00 00 00 29
4a30 : 00 00 00 00 00 00 00 00 31
4a38 : 00 00 00 00 1b 05 e0 00 96
4a40 : 00 00 00 00 80 00 00 00 49
4a48 : 00 00 80 00 00 00 00 00 69
4a50 : 80 00 00 00 00 00 80 00 d3
4a58 : 00 00 00 00 80 3f f0 00 1e
4a60 : 00 00 81 ff f0 00 00 00 d0
4a68 : 87 c0 00 00 00 00 9e 00 ca
4a70 : 00 00 00 00 b8 00 00 00 fc
4a78 : 00 00 e0 00 00 00 00 00 b1
4a80 : c0 00 00 00 00 00 80 00 43
4a88 : 00 00 00 00 00 00 00 00 89
4a90 : 00 00 00 00 00 00 00 00 91
4a98 : 00 00 00 00 00 00 00 00 99
4aa0 : 00 00 00 00 00 00 00 00 a1
4aa8 : 00 00 00 00 00 00 19 05 17
4ab0 : 1e 07 80 00 00 00 3f 08 7f
4ab8 : 40 00 00 00 63 90 20 00 34
4ac0 : 00 00 41 a0 20 00 00 00 27
4ac8 : c0 e0 10 00 00 00 80 c0 80
4ad0 : 10 00 00 00 80 60 10 00 2c
4ad8 : 00 00 80 60 10 00 00 00 06
4ae0 : 80 60 30 00 00 00 40 b0 ff
4ae8 : 20 00 00 00 41 38 60 00 60
4af0 : 00 00 22 1f c0 00 00 00 69
4af8 : 1c 0f 80 00 00 00 00 00 bc
4b00 : 00 00 00 00 00 00 00 00 01
4b08 : 00 00 00 00 00 00 00 00 09
4b10 : 00 00 00 00 00 00 00 00 11
4b18 : 00 00 00 00 00 00 00 00 19
4b20 : 19 05 0f 01 c0 00 00 00 ac
4b28 : 3f c3 e0 00 00 00 70 e3 0b
4b30 : 20 00 00 00 40 20 30 00 16
4b38 : 00 00 c0 30 10 00 00 00 70
4b40 : 80 10 10 00 00 00 80 10 ef
4b48 : 10 00 00 00 80 10 10 00 21
4b50 : 00 00 80 10 10 00 00 00 74
4b58 : c0 30 30 00 00 00 40 20 7e
4b60 : 20 00 00 00 70 e0 e0 00 12
4b68 : 00 00 3f ff c0 00 00 00 44
4b70 : 0f ff 00 00 00 00 00 00 7f
4b78 : 00 00 00 00 00 00 00 00 79
4b80 : 00 00 00 00 00 00 00 00 81
4b88 : 00 00 00 00 00 00 00 00 89
4b90 : 00 00 1b 05 00 20 20 00 79
4b98 : 00 00 00 70 70 00 00 00 ae
4ba0 : 00 20 20 00 00 00 00 00 b9
4ba8 : 00 00 00 00 00 00 00 00 a9
4bb0 : 00 00 00 00 00 00 00 00 b1
4bb8 : 00 00 00 00 00 00 00 00 b9
4bc0 : 00 00 00 00 00 00 00 00 c1
4bc8 : 00 00 00 00 00 00 00 00 c9
4bd0 : 00 00 00 00 00 00 00 00 d1
4bd8 : 00 00 00 00 00 00 00 00 d9
4be0 : 00 00 00 00 00 00 00 00 e1
4be8 : 00 00 00 00 00 00 00 00 e9
4bf0 : 00 00 00 00 00 00 00 00 f1
4bf8 : 00 00 00 00 00 00 00 00 f9
4c00 : 00 00 00 05 05 0a 00 20 82
4c08 : 20 00 00 00 00 70 72 00 76
4c10 : 00 04 00 20 38 00 00 00 9a
4c18 : 00 00 00 00 00 00 00 00 19
4c20 : 00 00 00 00 00 00 00 00 21
4c28 : 00 00 00 00 00 00 00 00 29
4c30 : 00 00 00 00 00 00 00 00 31
4c38 : 00 00 00 00 00 00 00 00 39
4c40 : 00 00 00 00 00 00 00 00 41
4c48 : 00 00 00 00 00 00 00 00 49
4c50 : 00 00 00 00 00 00 00 00 51
4c58 : 00 00 00 00 00 00 00 00 59
4c60 : 00 00 00 00 00 00 00 00 61
4c68 : 00 00 00 00 00 00 00 00 69
4c70 : 00 00 00 00 00 05 05 0a c1
4c78 : 00 00 00 00 00 00 00 00 79
4c80 : 00 00 00 00 00 00 00 00 81
4c88 : 00 00 00 00 00 00 00 00 89
4c90 : 00 00 00 00 00 00 00 00 91
4c98 : 00 00 00 00 00 00 00 00 99
4ca0 : 00 00 00 00 00 00 00 00 a1
4ca8 : 00 00 00 00 00 00 00 00 a9
4cb0 : 00 00 00 00 00 00 00 00 b1
4cb8 : 00 00 00 00 00 00 00 00 b9
4cc0 : 00 00 00 00 00 00 00 00 c1
4cc8 : 00 00 00 00 00 00 00 00 c9
4cd0 : 00 00 00 00 00 00 00 00 d1
4cd8 : 00 00 00 00 00 00 00 00 d9
4ce0 : 00 00 00 00 00 00 00 00 e1
4ce8 : 00 00 00 44 00 00 00 00 71
4cf0 : 00 22 00 00 00 00 00 22 46
4cf8 : 00 00 00 00 00 22 00 00 0a
4d00 : 00 00 00 66 00 00 00 00 cd
4d08 : 00 66 00 00 00 00 00 44 c4
4d10 : 00 00 00 00 00 44 00 00 33
4d18 : 00 00 00 22 00 00 00 00 5d
4d20 : 00 00 00 00 00 00 00 00 21
4d28 : 00 00 00 00 00 00 00 00 29
4d30 : 00 00 00 00 00 00 00 00 31
4d38 : 00 00 00 00 00 00 00 00 39
4d40 : 00 00 00 00 00 00 00 00 41
4d48 : 00 00 00 00 00 00 00 00 49
4d50 : 00 00 00 00 00 00 00 00 51
4d58 : 00 00 11 05 00 00 00 00 3d
4d60 : 00 00 00 00 00 00 00 00 61
4d68 : 00 00 00 00 00 00 00 00 69
4d70 : 00 00 00 00 00 00 00 00 71
4d78 : 00 00 00 00 00 00 00 00 79
4d80 : 00 00 00 00 00 00 00 00 81
4d88 : 00 00 00 00 00 00 00 00 89
4d90 : 00 00 00 00 00 00 00 00 91
4d98 : 00 00 00 00 00 00 00 00 99
4da0 : 00 00 00 00 00 00 00 00 a1
4da8 : 00 00 00 00 00 00 00 00 a9
4db0 : 00 00 00 00 00 00 00 00 b1
4db8 : 00 00 00 00 00 00 00 00 b9
4dc0 : 00 00 00 00 00 00 00 00 c1
4dc8 : 00 00 00 00 00 00 18 00 29
4dd0 : 00 00 00 00 3c 00 00 00 94
4dd8 : 00 00 64 1c 00 00 00 00 75
4de0 : 40 22 00 00 00 00 c0 62 f9
4de8 : 00 00 00 00 80 41 20 00 7b
4df0 : 00 00 80 41 70 00 00 00 40
4df8 : 80 41 20 00 00 00 c0 c2 aa
4e00 : 00 00 00 00 40 84 00 00 29
4e08 : 00 00 61 80 00 00 00 00 71
4e10 : 3f 00 00 00 00 00 1e 00 c8
4e18 : 00 00 00 00 00 00 00 00 19
4e20 : 00 00 00 00 00 00 00 00 21
4e28 : 00 00 00 00 00 00 00 00 29
4e30 : 00 00 00 00 00 00 00 00 31
4e38 : 00 00 00 00 00 00 19 05 a7
4e40 : 00 00 00 00 00 00 00 00 41
4e48 : 00 00 00 00 00 00 00 00 49
4e50 : 00 00 00 00 00 00 00 00 51
4e58 : 00 00 00 00 00 00 00 00 59
4e60 : 00 00 00 00 00 00 00 00 61
4e68 : 00 00 00 00 00 00 00 00 69
4e70 : 00 00 00 00 00 00 00 00 71
4e78 : 00 00 00 00 00 00 00 00 79
4e80 : 00 00 00 00 00 00 00 00 81
4e88 : 00 00 00 00 00 00 00 00 89
4e90 : 00 00 00 00 00 00 00 00 91
4e98 : 00 00 00 00 00 00 00 00 99
4ea0 : 00 00 00 00 00 00 00 00 a1
4ea8 : 00 00 00 00 00 00 00 00 a9
4eb0 : 00 00 38 11 00 00 00 00 e1
4eb8 : 4e 21 00 00 00 80 87 c1 3d
4ec0 : 00 00 02 80 83 81 00 00 95
4ec8 : 02 c0 c0 05 00 00 00 c0 7d
4ed0 : 70 18 20 00 00 d0 3f f0 ba
4ed8 : 20 00 00 40 0f c0 20 00 78
4ee0 : 00 00 00 00 00 00 00 40 61
4ee8 : 00 00 00 00 00 00 80 00 eb
4ef0 : 80 40 00 00 3f ff e0 00 08
4ef8 : 00 00 3f ff f0 00 00 00 d7
4f00 : 40 00 10 00 00 00 80 00 47
4f08 : 20 00 00 00 80 00 00 00 31
4f10 : 00 00 00 00 00 00 00 00 11
4f18 : 00 00 00 00 00 00 00 00 19
4f20 : 00 00 1d 05 78 00 10 00 d0
4f28 : 00 00 c7 00 20 00 00 00 1c
4f30 : 80 00 20 00 00 00 c0 01 be
4f38 : 60 00 00 00 7f fe 60 00 0a
4f40 : 00 00 3f fc 60 00 00 00 b6
4f48 : 60 80 60 00 00 00 c0 80 05
4f50 : 20 00 00 00 80 80 30 00 3d
4f58 : 00 00 80 80 10 00 00 00 8a
4f60 : c0 80 10 00 00 00 40 c0 e7
4f68 : 10 00 00 00 70 c0 20 08 16
4f70 : 00 00 31 70 60 08 00 80 12
4f78 : 05 3f 40 00 00 80 02 0f 57
4f80 : 00 00 00 00 00 00 00 00 81
4f88 : 00 00 00 00 00 00 00 00 89
4f90 : 00 00 00 00 1f 05 07 fe c5
4f98 : 00 00 00 00 1f ff 80 00 8c
4fa0 : 00 00 70 01 c0 00 00 00 e9
4fa8 : c0 00 60 00 00 00 80 00 83
4fb0 : 20 00 06 00 4f 80 30 00 0c
4fb8 : 01 00 3f e0 30 00 00 80 a9
4fc0 : 20 f0 10 00 00 80 40 78 52
4fc8 : 10 00 01 00 80 1e 10 00 52
4fd0 : 00 00 c0 00 30 00 00 00 04
4fd8 : 60 00 20 00 00 00 60 00 c2
4fe0 : 60 00 00 00 40 00 40 00 46
4fe8 : 00 00 80 00 80 00 00 00 11
4ff0 : 00 00 00 00 00 00 00 00 f1
4ff8 : 00 00 00 00 00 00 00 00 f9
5000 : 00 00 00 00 00 00 1d 05 7f
5008 : 60 00 10 00 00 00 40 00 6e
5010 : 20 00 0c 00 8f 02 60 00 be
5018 : 00 00 9f 81 60 00 00 00 36
5020 : a1 c1 60 00 00 00 c0 fe bb
5028 : 30 00 00 00 80 3c 10 00 83
5030 : 00 00 80 00 10 00 00 00 52
5038 : 80 00 10 00 00 00 c0 00 c0
5040 : 30 00 00 00 40 00 20 00 f5
5048 : 00 00 60 00 60 00 00 00 67
5050 : 30 00 c0 0c 03 00 13 fc a8
5058 : 80 0c 03 00 03 fc 00 00 b7
```

Listing 2. (Fortsetzung)

```
5060 : 00 00 00 00 00 00 00 00  61
5068 : 00 00 00 00 00 00 00 00  69
5070 : 00 00 00 00 00 00 00 00  71
5078 : 1d 05 07 fe 00 00 00 00  ba
5080 : 1f ff 80 00 00 00 70 01  83
5088 : c0 00 00 00 c0 00 60 00  d6
5090 : 00 00 80 00 20 00 06 00  cb
5098 : 4f 80 30 00 01 00 3f e0  02
50a0 : 30 00 00 80 20 f0 10 00  aa
50a8 : 00 80 40 78 10 00 01 00  0d
50b0 : 80 9e 10 00 00 00 c1 80  8c
50b8 : 30 00 00 00 61 80 20 00  83
50c0 : 00 00 60 c0 60 00 00 00  f7
50c8 : 40 40 40 00 00 00 80 00  3b
50d0 : 80 00 00 00 00 00 00 00  51
50d8 : 00 00 00 00 00 00 00 00  d9
50e0 : 00 00 00 00 00 00 00 00  e1
50e8 : 00 00 1d 05 38 00 38 00  35
50f0 : 00 00 7e 20 fc 00 00 00  64
50f8 : c3 13 c6 00 00 00 80 8f  18
5100 : 02 00 00 00 80 00 03 00  17
5108 : 00 00 c0 00 01 00 00 00  49
5110 : 40 00 01 00 00 00 67 80  2f
5118 : 01 00 00 00 7f f0 01 00  9d
5120 : 00 00 20 7e 02 00 00 00  18
5128 : 60 0f c2 00 00 00 60 11  64
5130 : fc 00 00 00 40 30 30 00  73
5138 : 00 00 40 30 00 00 00 00  4f
5140 : 80 10 00 00 00 00 80 08  db
5148 : 00 00 00 00 00 00 00 00  49
5150 : 00 00 00 00 00 00 00 00  51
5158 : 00 00 00 00 1f 05 07 fe  8d
5160 : 00 00 00 00 1f ff 80 00  54
5168 : 00 00 70 01 c0 00 00 00  b1
5170 : c0 00 60 00 00 00 80 00  4b
5178 : 20 00 03 00 47 c0 b0 00  96
5180 : 00 00 2f e0 b0 00 00 00  73
5188 : 10 f1 90 00 00 00 20 7f  34
5190 : 10 00 00 00 40 be 10 00  db
5198 : 00 00 c1 00 30 00 00 00  0c
51a0 : 61 00 20 00 00 00 61 80  90
51a8 : 60 00 00 00 40 e1 c0 00  1f
51b0 : 00 00 80 7f 80 00 00 00  c8
51b8 : 00 1e 00 00 00 00 00 00  c8
51c0 : 00 00 00 00 00 00 00 00  c1
51c8 : 00 00 00 00 00 00 1f 05  4f
51d0 : 00 00 60 00 00 00 0f c0  a6
51d8 : 90 00 00 00 1f f0 10 00  22
51e0 : 00 00 20 7e 10 00 00 00  b9
51e8 : 40 2f e0 00 00 00 80 43  81
51f0 : c0 00 00 00 c0 40 0c 00  ef
51f8 : 00 00 c0 40 12 00 00 00  52
5200 : 60 80 21 00 00 00 60 80  6b
5208 : 71 00 00 00 61 80 38 00  74
5210 : 00 00 41 80 1c 00 00 00  33
5218 : 80 e0 08 00 00 00 00 7c  03
5220 : 10 00 00 00 00 3f e0 00  ae
5228 : 00 00 00 07 c0 00 00 00  15
5230 : 00 00 00 00 00 00 00 00  31
5238 : 00 00 00 00 00 00 00 00  39
5240 : 1f 05 38 01 80 00 00 00  18
5248 : 7c 01 00 00 00 00 c2 01  52
5250 : 80 00 00 00 80 00 e0 00  5c
5258 : 00 00 c0 00 70 00 00 00  90
5260 : c0 00 30 00 00 00 63 80  bb
5268 : 10 00 00 00 65 c0 10 00  15
5270 : 00 00 28 e0 20 00 00 00  99
5278 : 28 70 20 00 00 00 64 3f  f1
5280 : c0 00 00 00 40 0f 00 00  bd
5288 : 00 00 40 00 00 00 00 00  99
5290 : 80 00 00 00 00 00 00 00  11
5298 : 00 00 00 00 00 00 00 00  99
52a0 : 00 00 00 00 00 00 00 00  a1
52a8 : 00 00 00 00 00 00 00 00  a9
```

```
52b0 : 00 00 1b 05 38 00 0c 00  cc
52b8 : 00 00 7c 00 12 00 00 00  f9
52c0 : c4 00 10 00 00 00 80 00  8b
52c8 : 18 00 00 00 c0 00 0c 00  1d
52d0 : 00 00 c1 e0 0e 00 00 00  3e
52d8 : 63 f0 07 00 00 00 64 38  77
52e0 : 03 00 00 00 28 1c 02 00  4f
52e8 : 00 00 28 0f 02 00 00 00  f5
52f0 : 64 07 c4 00 00 00 40 01  0c
52f8 : f8 00 00 00 40 00 60 00  76
5300 : 00 00 80 00 00 00 00 00  21
5308 : 00 00 00 00 00 00 00 00  09
5310 : 00 00 00 00 00 00 00 00  11
5318 : 00 00 00 00 00 00 00 00  19
5320 : 00 00 00 00 1b 05 00 00  fa
5328 : 60 00 00 00 0f 00 90 00  bc
5330 : 00 00 1f c0 10 00 00 00  11
5338 : 38 f8 20 00 00 00 63 3f  01
5340 : c0 00 00 00 4c 27 80 00  00
5348 : 00 00 c8 20 00 00 00 00  7f
5350 : 90 20 00 00 00 00 90 20  73
5358 : 00 00 00 00 90 20 00 00  63
5360 : 00 00 90 60 00 00 00 00  91
5368 : 98 7c 00 00 00 00 ce 5f  38
5370 : c0 00 00 00 47 43 e0 00  43
5378 : 00 00 60 80 10 00 00 00  a2
5380 : 38 00 20 00 00 00 0f 00  fd
5388 : 00 00 00 00 00 00 00 00  89
5390 : 00 00 00 00 00 00 21 05  1f
5398 : 00 00 10 00 00 00 07 80  ba
53a0 : 60 00 00 00 1f e1 a0 00  84
53a8 : 00 00 39 ff 60 00 00 00  fd
53b0 : 60 3c 40 00 00 00 40 00  40
53b8 : c0 00 00 00 c0 00 c0 00  88
53c0 : 00 00 80 00 60 00 00 00  e7
53c8 : 80 00 70 00 00 00 80 00  67
53d0 : 30 00 00 00 c0 00 30 00  cd
53d8 : 00 00 40 e0 20 00 00 00  07
53e0 : 63 90 60 00 00 00 3f 10  c1
53e8 : 40 00 00 00 0e 20 80 00  0c
53f0 : 00 00 00 00 00 00 00 00  f1
53f8 : 00 00 00 00 00 00 00 00  f9
5400 : 00 00 00 00 00 00 00 00  01
5408 : 1d 05 78 00 00 00 00 00  c6
5410 : c7 00 10 00 00 00 80 00  de
5418 : 20 00 00 00 c0 00 10 00  85
5420 : 00 00 3f ff e0 00 00 00  fe
5428 : 7f ff c0 00 00 00 80 00  d9
5430 : 00 00 00 10 80 00 00 00  3b
5438 : 00 20 80 00 00 40 00 10  8b
5440 : 00 00 00 3f ff e0 00 00  2f
5448 : 00 7f ff c0 00 00 00 80  21
5450 : 00 00 00 00 00 80 00 00  55
5458 : 00 00 00 80 00 00 40 00  6a
5460 : 00 00 00 00 78 7f c0 00  e7
5468 : 00 00 1c ff e0 00 00 00  7d
5470 : 05 00 10 00 00 00 02 00  82
5478 : 20 00 25 05 78 00 00 00  0a
5480 : 00 00 c7 00 10 00 00 00  73
5488 : 80 00 20 00 00 00 80 00  13
5490 : 10 40 00 00 3f ff e0 00  38
5498 : 00 00 3f ff c0 00 00 00  74
54a0 : 40 00 00 00 00 00 80 00  e3
54a8 : 00 00 00 00 80 00 00 00  b1
54b0 : 00 00 80 00 00 00 00 00  d1
54b8 : c0 00 00 00 00 00 78 7f  59
54c0 : c0 00 00 00 3c ff e0 00  c8
54c8 : 00 00 05 00 10 00 00 00  0b
54d0 : 02 00 20 00 00 00 00 00  db
54d8 : 00 00 00 00 00 00 00 00  d9
54e0 : 00 00 00 00 00 00 00 00  e1
54e8 : 00 00 00 00 1d 05 0f 80  20
54f0 : 10 00 00 00 3f e0 30 00  bc
54f8 : 00 00 70 70 20 00 00 00  25
```

```
5500 : c0 38 e0 00 00 00 80 1f  55
5508 : 20 00 00 00 80 0e 30 00  62
5510 : 00 00 c0 00 30 00 00 00  44
5518 : c0 00 10 00 00 00 c0 00  e0
5520 : 10 00 00 00 c0 00 10 00  7d
5528 : 00 00 c0 00 10 00 00 00  5a
5530 : 40 00 30 00 00 00 60 00  fe
5538 : 20 00 00 00 38 01 e0 00  68
5540 : 00 00 1f ff 00 00 00 00  08
5548 : 07 fe 00 00 00 00 00 00  cf
5550 : 00 00 00 00 00 00 00 00  51
5558 : 00 00 00 00 00 00 1f 05  df
5560 : 78 00 30 00 00 00 c7 00  04
5568 : 40 00 00 00 80 00 40 00  b2
5570 : 00 00 80 00 60 40 00 00  99
5578 : 3f ff ff 00 00 00 3f ff  b4
5580 : ff 00 00 00 40 00 30 00  44
5588 : 00 00 80 00 30 00 00 00  ac
5590 : 80 00 10 00 00 00 80 00  17
5598 : 10 00 00 00 c0 00 30 00  75
55a0 : 00 00 78 40 20 00 00 00  c9
55a8 : 1c f0 e0 00 00 00 05 1f  c7
55b0 : c0 00 00 00 02 07 00 00  c9
55b8 : 00 00 00 00 00 00 00 00  b9
55c0 : 00 00 00 00 00 00 00 00  c1
55c8 : 00 00 00 00 00 00 00 00  c9
55d0 : 1d 05 0f 80 10 00 00 00  45
55d8 : 3f e0 30 00 00 00 70 70  36
55e0 : 20 00 00 00 c0 38 e0 00  52
55e8 : 00 00 80 1f 20 00 00 00  ee
55f0 : 80 0e 30 00 00 00 c0 00  87
55f8 : 30 00 00 00 c0 00 10 00  75
5600 : 00 00 c0 00 10 00 00 00  32
5608 : c0 00 10 00 00 00 c0 00  d0
5610 : 10 00 00 00 40 00 30 00  e5
5618 : 00 00 60 00 60 00 00 00  37
5620 : 38 01 c0 00 00 00 1f ff  85
5628 : 60 00 00 00 07 fe 30 00  b2
5630 : 00 00 00 00 10 00 00 00  32
5638 : 00 00 20 00 00 00 00 00  41
5640 : 40 00 25 05 78 00 00 00  f2
5648 : 00 00 ce 00 00 00 00 00  fc
5650 : 81 80 20 00 00 00 80 00  1c
5658 : 10 40 00 00 3f ff e0 00  00
5660 : 00 00 3f ff c0 00 00 00  3c
5668 : 60 40 00 00 00 00 c0 40  6c
5670 : 00 00 00 00 80 40 00 00  7b
5678 : 00 00 80 80 00 00 00 00  a9
5680 : 80 80 00 00 00 00 c0 c0  c5
5688 : 00 00 00 00 40 e0 00 00  94
5690 : 00 00 71 7f e0 08 00 00  2b
5698 : 31 3f f0 08 00 00 05 00  ba
56a0 : 10 00 00 00 02 00 20 00  51
56a8 : 00 00 00 00 00 00 00 00  a9
56b0 : 00 00 00 00 21 05 07 fe  05
56b8 : 00 00 00 00 1f ff 80 00  ac
56c0 : 00 00 38 01 c0 00 00 00  fb
56c8 : 60 00 60 00 00 00 40 30  a2
56d0 : 20 00 00 00 c0 48 30 00  00
56d8 : 00 00 c0 88 30 00 00 00  1d
56e0 : c0 88 10 00 00 00 40 98  1b
56e8 : 10 00 00 00 60 10 10 00  bf
56f0 : 00 00 20 20 30 00 00 00  00
56f8 : 20 20 20 00 00 00 20 30  11
5700 : 60 00 00 00 60 1f c0 00  62
5708 : 00 00 40 0f 00 00 00 00  fa
5710 : 80 00 00 00 00 00 00 00  91
5718 : 00 00 00 00 00 00 00 00  19
5720 : 00 00 00 00 00 00 1f 05  a7
5728 : 38 00 10 00 00 00 44 00  76
5730 : 20 00 00 00 82 00 60 00  fa
5738 : 00 00 c0 00 60 00 00 00  6f
5740 : c0 02 60 00 00 00 67 81  ba
5748 : 30 00 00 00 69 c0 b0 00  d8
```

```
5750 : 00 00 30 e0 90 00 00 00  82
5758 : 20 7f 10 00 00 00 20 1c  f5
5760 : 30 00 00 00 60 00 20 00  17
5768 : 00 00 40 00 20 00 00 00  7b
5770 : 80 00 40 00 00 00 80 00  03
5778 : 40 00 00 00 00 00 00 00  b9
5780 : 00 00 00 00 00 00 00 00  81
5788 : 00 00 00 00 00 00 00 00  89
5790 : 00 00 00 00 00 00 00 00  91
5798 : 1b 05 40 00 40 00 00 00  4a
57a0 : 80 00 c0 00 00 00 ff ff  50
57a8 : e0 00 00 00 7f ff 60 00  02
57b0 : 00 00 00 00 30 00 00 00  b4
57b8 : 00 00 10 00 00 00 00 00  bd
57c0 : 10 00 00 00 00 00 10 00  11
57c8 : 00 00 00 00 10 00 00 00  ca
57d0 : 40 00 20 00 00 00 ff ff  18
57d8 : c0 00 00 00 7f ff e0 00  14
57e0 : 00 00 20 00 10 00 00 00  ea
57e8 : 00 00 20 00 00 00 00 00  f1
57f0 : 00 00 00 00 00 00 00 00  f1
57f8 : 00 00 00 00 00 00 00 00  f9
5800 : 00 00 00 00 00 00 00 00  01
5808 : 00 00 1b 05 78 00 10 00  38
5810 : 00 00 c7 00 20 00 00 00  04
5818 : 80 00 40 00 00 00 80 00  ab
5820 : 60 40 00 00 3f ff e0 00  18
5828 : 00 00 3f ff f0 00 00 00  07
5830 : 40 00 30 00 00 00 80 00  7f
5838 : 30 00 00 00 80 00 10 00  b1
5840 : 00 00 80 00 10 00 00 00  62
5848 : c0 00 30 00 00 00 78 70  d7
5850 : 60 00 00 00 1c fe 60 00  ec
5858 : 00 00 05 1f c0 00 00 00  8a
5860 : 02 03 80 00 00 00 00 00  04
5868 : 00 00 00 00 00 00 00 00  69
5870 : 00 00 00 00 00 00 00 00  71
5878 : 00 00 00 00 1d 05 78 00  54
5880 : 20 00 00 00 c7 00 40 00  1e
5888 : 00 00 80 00 40 00 00 00  ad
5890 : 80 01 e0 40 00 00 3f ff  ce
5898 : e0 00 00 00 3f ff b0 00  2f
58a0 : 00 00 40 00 10 00 00 00  b2
58a8 : 80 00 10 00 00 00 80 00  2f
58b0 : 10 00 00 20 40 00 00 3f  47
58b8 : ff e0 00 00 00 3f ff c0  a3
58c0 : 40 00 20 00 00 00 80 00  0b
58c8 : 10 00 00 00 80 00 10 00  21
58d0 : 00 00 c0 00 30 00 00 00  04
58d8 : 78 70 20 00 00 00 1c f8  f3
58e0 : 60 00 00 00 05 1f c0 00  8d
58e8 : 00 00 02 07 00 00 25 05  e8
58f0 : 40 00 00 00 00 00 80 00  33
58f8 : 20 00 00 00 80 20 10 00  62
5900 : 00 00 c0 40 10 00 00 00  3a
5908 : c0 40 10 00 00 00 7f ff  ea
5910 : e0 00 00 00 7f ff e0 00  6c
5918 : 00 00 80 40 30 00 00 00  44
5920 : 80 40 30 00 00 00 80 80  d0
5928 : 10 00 00 00 40 00 10 00  7d
5930 : 00 00 00 00 20 00 00 00  33
5938 : 00 00 00 00 00 00 00 00  39
5940 : 00 00 00 00 00 00 00 00  41
5948 : 00 00 00 00 00 00 00 00  49
5950 : 00 00 00 00 00 00 00 00  51
5958 : 00 00 00 00 00 00 00 00  59
5960 : 17 05 78 00 00 00 00 00  18
5968 : c7 00 04 00 00 00 80 00  33
5970 : 04 00 00 00 80 00 06 40  15
5978 : 00 00 3f ff c2 00 00 00  74
5980 : 3f ff e3 00 00 00 40 00  b9
5988 : 11 00 00 00 c0 00 21 00  2a
5990 : 00 00 80 00 01 00 00 00  c1
5998 : 80 00 03 00 00 00 80 00  db
59a0 : 02 00 00 00 c0 00 06 00  c7
59a8 : 00 08 70 00 14 0c 00 68  3b
59b0 : 33 ff 90 0c 00 60 03 ff  98
59b8 : 80 00 00 00 00 00 00 00  39
59c0 : 00 00 00 00 00 00 00 00  c1
59c8 : 00 00 00 00 00 00 00 00  c9
59d0 : 00 00 1d 05 00 00 1c 0c  41
59d8 : 00 00 10 08 30 00 00 00  e1
59e0 : 20 10 30 00 00 00 41 d0  bb
59e8 : 1c 00 00 00 81 20 06 00  36
59f0 : 00 00 80 20 01 00 00 00  25
59f8 : 80 20 01 00 00 00 c0 70  ad
5a00 : 01 00 00 00 40 50 01 00  8c
5a08 : 00 00 60 d8 03 00 00 00  6c
5a10 : 31 88 02 0e 00 00 11 0c  24
5a18 : 06 0e 03 18 00 04 e4 00  9d
5a20 : 03 18 00 00 e0 00 00 00  3e
5a28 : 00 00 00 00 00 00 00 00  29
5a30 : 00 00 00 00 00 00 00 00  31
5a38 : 00 00 00 00 00 00 00 00  39
5a40 : 00 00 00 00 19 05 07 02  1a
5a48 : 00 00 00 00 0f c4 00 00  60
5a50 : 00 00 18 78 80 00 00 00  6e
5a58 : 90 01 80 40 00 00 98 02  f7
5a60 : c0 00 00 00 0f fc 60 00  7b
5a68 : 00 00 07 f8 30 00 00 00  4c
5a70 : 00 00 30 00 00 00 10 00  bd
5a78 : 20 00 00 00 88 00 40 40  a3
5a80 : 00 00 87 ff e0 00 00 00  70
5a88 : 0f ff f0 00 00 00 10 00  13
5a90 : 20 00 00 00 20 00 40 00  b4
5a98 : 00 00 20 00 00 00 00 00  a1
5aa0 : 00 00 00 00 00 00 00 00  a1
5aa8 : 00 00 00 00 00 00 00 00  a9
5ab0 : 00 00 00 00 00 00 1d 05  2f
5ab8 : 03 80 10 00 00 00 07 e1  df
5ac0 : 30 00 00 00 0c 70 a0 00  b7
5ac8 : 00 00 98 38 e0 40 00 00  06
5ad0 : 90 1d 60 00 00 00 10 0e  63
5ad8 : 20 00 00 00 18 00 30 00  3b
5ae0 : 00 00 18 00 10 00 00 00  e8
5ae8 : 18 00 10 00 00 00 98 00  67
5af0 : 10 40 00 00 98 00 10 00  ea
5af8 : 00 00 08 00 30 00 00 00  fe
5b00 : 0c 00 20 00 00 00 07 00  31
5b08 : e0 00 00 00 03 ff c0 00  1c
5b10 : 00 00 00 ff 00 00 00 00  10
5b18 : 00 00 00 00 00 00 00 00  19
5b20 : 00 00 00 00 00 00 00 00  21
5b28 : 1f 05 08 00 40 00 00 00  d0
5b30 : 10 00 c0 00 00 00 9f ff  ef
5b38 : e0 40 00 00 8f ff 60 00  b3
5b40 : 00 00 00 00 30 00 00 00  44
5b48 : 00 00 10 00 00 00 00 00  4d
5b50 : 10 00 00 00 00 00 10 00  a1
5b58 : 00 00 00 00 10 80 00 00  5e
5b60 : 48 00 20 80 00 00 1f ff  3d
5b68 : c0 00 00 00 0f ff e0 00  9d
5b70 : 00 00 04 00 10 00 00 00  73
5b78 : 00 00 20 00 00 00 00 00  81
5b80 : 00 00 00 00 00 00 00 00  81
5b88 : 00 00 00 00 00 00 00 00  89
5b90 : 00 00 00 00 00 00 00 00  91
5b98 : 00 00 1b 05 7f fe 00 00  f0
5ba0 : 00 00 e3 ff ff 00 00 00  99
5ba8 : c0 82 04 00 00 00 41 82  b5
5bb0 : 02 00 00 00 61 04 02 00  f1
5bb8 : 00 00 61 84 06 00 00 00  02
5bc0 : 40 fb fc 00 00 00 80 71  22
5bc8 : f8 00 00 00 00 00 00 00  c1
5bd0 : 00 00 00 00 00 00 00 00  d1
5bd8 : 00 00 00 00 00 00 00 00  d9
5be0 : 00 00 00 00 00 00 00 00  e1
5be8 : 00 00 00 00 00 00 00 00  e9
5bf0 : 00 00 00 00 00 00 00 00  f1
5bf8 : 00 00 00 00 00 00 00 00  f9
5c00 : 00 00 00 00 00 00 00 00  01
5c08 : 00 00 00 03 0f 03 00 00  72
5c10 : 00 00 00 00 00 00 00 00  11
5c18 : 00 00 00 00 00 00 00 00  19
5c20 : 00 00 00 00 00 00 00 00  21
5c28 : 00 00 00 00 00 00 00 00  29
5c30 : 00 00 00 00 00 00 00 00  31
5c38 : 00 00 00 00 00 00 00 00  39
5c40 : 00 00 00 00 00 00 00 00  41
5c48 : 00 00 00 00 00 00 00 00  49
5c50 : 00 00 00 00 00 00 00 00  51
5c58 : 00 00 00 00 00 00 00 00  59
5c60 : 00 00 00 00 00 00 00 00  61
5c68 : 00 00 00 00 00 00 00 00  69
5c70 : 00 00 00 00 00 00 00 00  71
5c78 : 00 00 00 00 00 00 00 00  79
5c80 : 00 00 00 00 00 00 00 00  81
5c88 : 00 00 00 00 00 00 00 00  89
5c90 : 00 00 00 00 00 00 00 00  91
5c98 : 00 00 00 00 00 00 00 00  99
5ca0 : 00 00 00 00 00 00 00 00  a1
5ca8 : 00 00 00 00 00 00 00 00  a9
5cb0 : 00 00 00 00 00 00 00 00  b1
5cb8 : 00 00 00 00 00 00 00 00  b9
5cc0 : 00 00 00 00 00 00 00 00  c1
5cc8 : 00 00 00 00 00 00 00 00  c9
5cd0 : 00 00 00 00 00 00 00 00  d1
5cd8 : 00 00 00 00 00 00 00 00  d9
5ce0 : 00 00 00 00 00 00 00 00  e1
5ce8 : 00 00 00 00 00 00 00 00  e9
5cf0 : 00 00 00 1f 80 00 60 60  1f
5cf8 : 00 1f 80 00 e0 70 00 00  3a
5d00 : 00 01 00 30 00 00 00 03  8d
5d08 : 00 10 00 00 00 03 00 20  69
5d10 : 00 00 00 03 80 20 00 00  7a
5d18 : 00 01 80 40 00 00 00 01  c3
5d20 : ff e0 00 00 00 00 ff f0  71
5d28 : 00 00 00 00 80 10 00 00  b1
5d30 : 00 00 00 20 00 00 00 00  35
5d38 : 00 00 00 00 00 00 00 00  39
5d40 : 00 00 00 00 00 00 00 00  41
5d48 : 00 00 00 00 00 00 00 00  49
5d50 : 00 00 00 00 00 00 00 00  51
5d58 : 00 00 00 00 00 00 00 00  59
5d60 : 00 00 16 05 80 00 80 40  11
5d68 : 00 00 3f ff 80 00 00 40  c1
5d70 : 3f ff 80 00 00 40 40 80  d3
5d78 : 20 00 00 40 80 00 20 01  2b
5d80 : 00 10 80 00 20 00 00 10  cb
5d88 : 02 00 00 00 00 00 03 00  97
5d90 : 20 00 00 00 01 80 00 00  c5
5d98 : 61 c0 00 9e 00 00 61 80  b4
5da0 : 00 1e 00 00 00 00 00 00  b0
5da8 : 00 00 00 00 00 00 00 00  a9
5db0 : 00 00 00 00 00 00 00 00  b1
5db8 : 00 00 00 00 00 00 00 00  b9
5dc0 : 00 00 00 00 00 00 00 00  c1
5dc8 : 00 00 00 00 00 00 00 00  c9
5dd0 : 00 00 00 00 15 05 00 1f  88
5dd8 : 00 00 60 c0 00 1f 00 00  01
5de0 : e0 e0 00 00 00 01 00 70  19
5de8 : 00 00 00 03 00 30 00 00  ca
5df0 : 00 03 80 00 00 00 20 01  15
5df8 : c0 00 00 00 00 00 80 40  3b
5e00 : 00 00 00 00 00 40 00 00  03
5e08 : 00 00 00 00 00 00 00 00  09
5e10 : 00 00 00 00 00 00 00 00  11
5e18 : 00 00 00 00 00 00 00 00  19
5e20 : 00 00 00 00 00 00 00 00  21
```

Listing 2. (Fortsetzung)

```
5e28 : 00 00 00 00 00 00 00 00  29
5e30 : 00 00 00 00 00 00 00 00  31
5e38 : 00 00 00 00 00 00 00 00  39
5e40 : 00 00 00 00 00 00 10 00  81
5e48 : 00 00 40 00 00 00 30 7f  18
5e50 : c0 00 00 20 78 ff c0 00  9f
5e58 : 00 20 cd 00 10 00 00 20  1d
5e60 : 86 00 10 00 00 00 83 00  f9
5e68 : 10 00 00 00 03 00 00 00  a9
5e70 : 00 20 01 c0 00 00 00 00  d9
5e78 : 00 ff c0 00 00 00 00 3f  26
5e80 : 80 00 00 00 00 00 00 00  01
5e88 : 00 00 00 00 00 00 00 00  89
5e90 : 00 00 00 00 00 00 00 00  91
5e98 : 00 00 00 00 00 00 00 00  99
5ea0 : 00 00 00 00 00 00 00 00  a1
5ea8 : 00 00 00 00 00 00 00 00  a9
5eb0 : 00 00 00 00 00 00 00 00  b1
5eb8 : 13 05 00 7f c0 00 00 00  4a
5ec0 : 00 ff e0 00 00 00 01 04  04
5ec8 : 70 00 00 00 03 0c 30 00  8a
5ed0 : 00 00 01 88 20 00 00 00  24
5ed8 : 00 d8 60 00 00 00 00 70  3d
5ee0 : 40 00 00 00 00 20 80 00  24
5ee8 : 00 00 00 00 00 00 00 00  e9
5ef0 : 00 00 00 00 00 00 00 00  f1
5ef8 : 00 00 00 00 00 00 00 00  f9
5f00 : 00 00 00 00 00 00 00 00  01
5f08 : 00 00 00 00 00 00 00 00  09
5f10 : 00 00 00 00 00 00 00 00  11
5f18 : 00 00 00 00 00 00 00 00  19
5f20 : 00 00 00 00 00 00 00 00  21
5f28 : 00 00 0f 05 01 00 00 00  9d
5f30 : 00 00 01 00 00 00 00 00  71
5f38 : 7f ff ff 00 00 00 81 ff  bd
5f40 : 00 00 00 00 c1 00 00 00  5d
5f48 : 00 00 61 00 00 00 00 00  a1
5f50 : 61 00 00 00 00 00 40 00  b3
5f58 : 00 00 00 00 80 00 00 00  61
5f60 : 00 00 00 00 00 00 00 00  61
5f68 : 00 00 00 00 00 00 00 00  69
5f70 : 00 00 00 00 00 00 00 00  71
5f78 : 00 00 00 00 00 00 00 00  79
5f80 : 00 00 00 00 00 00 00 00  81
5f88 : 00 00 00 00 00 00 00 00  89
5f90 : 00 00 00 00 00 00 00 00  91
5f98 : 00 00 00 00 11 05 00 7f  d1
5fa0 : c6 00 00 00 00 ff c9 00  8e
5fa8 : 00 20 01 00 45 00 00 20  8d
5fb0 : 03 00 11 00 00 20 03 80  06
5fb8 : 11 00 00 00 01 80 11 00  22
5fc0 : 00 00 01 c0 21 00 00 00  2b
5fc8 : 00 ff c3 00 00 00 00 ff  b9
5fd0 : fe 00 00 00 00 40 1c 00  41
5fd8 : 00 00 00 00 00 00 00 00  d9
5fe0 : 00 00 00 00 00 00 00 00  e1
5fe8 : 00 00 00 00 00 00 00 00  e9
5ff0 : 00 00 00 00 00 00 00 00  f1
5ff8 : 00 00 00 00 00 00 00 00  f9
6000 : 00 00 00 00 00 00 00 00  01
6008 : 00 00 00 00 00 00 13 05  5f
6010 : 80 00 20 40 00 00 3f ff  9d
6018 : e1 00 00 00 3f ff f1 00  b5
6020 : 00 00 40 80 11 00 00 00  52
6028 : 81 00 23 00 00 00 81 00  78
6030 : 02 00 00 00 02 00 06 00  6b
6038 : 00 00 03 c0 1c 00 00 00  d3
6040 : 01 ff f8 00 00 00 00 7f  7e
6048 : e0 00 00 00 00 00 00 00  29
6050 : 00 00 00 00 00 00 00 00  51
6058 : 00 00 00 00 00 00 00 00  59
6060 : 00 00 00 00 00 00 00 00  61
6068 : 00 00 00 00 00 00 00 00  69
6070 : 00 00 00 00 00 00 00 00  71
6078 : 00 00 00 00 00 00 00 00  79
6080 : 13 05 01 00 00 00 00 00  56
6088 : 02 00 20 00 00 00 03 ff  9f
6090 : e0 20 00 00 51 ff f0 20  9a
6098 : 00 00 40 00 10 00 00 00  aa
60a0 : 00 00 20 00 00 00 00 00  a9
60a8 : 00 00 00 00 00 00 00 00  a9
60b0 : 00 00 00 00 00 00 00 00  b1
60b8 : 00 00 00 00 00 00 00 00  b9
60c0 : 00 00 00 00 00 00 00 00  c1
60c8 : 00 00 00 00 00 00 00 00  c9
60d0 : 00 00 00 00 00 00 00 00  d1
60d8 : 00 00 00 00 00 00 00 00  d9
60e0 : 00 00 00 00 00 00 00 00  e1
60e8 : 00 00 00 00 00 00 00 00  e9
60f0 : 00 00 0b 05 00 00 01 00  58
60f8 : 00 00 01 00 01 00 00 00  49
6100 : 02 00 02 00 00 00 23 ff  10
6108 : fc 50 00 00 21 ff f8 40  a3
6110 : 00 04 00 80 02 00 00 00  43
6118 : 00 00 00 00 00 00 00 00  19
6120 : 00 00 00 00 00 00 00 00  21
6128 : 00 00 00 00 00 00 00 00  29
6130 : 00 00 00 00 00 00 00 00  31
6138 : 00 00 00 00 00 00 00 00  39
6140 : 00 00 00 00 00 00 00 00  41
6148 : 00 00 00 00 00 00 00 00  49
6150 : 00 00 00 00 00 00 00 00  51
6158 : 00 00 00 00 00 00 00 00  59
6160 : 00 00 00 00 0b 05 80 00  3b
6168 : 20 40 00 00 3f ff e0 00  20
6170 : 00 00 3f ff f0 00 00 00  4f
6178 : 40 82 10 00 00 00 81 04  0c
6180 : 20 00 00 00 82 08 00 00  09
6188 : 00 00 02 18 00 00 00 00  0c
6190 : 03 2c c0 00 00 00 01 cf  7d
6198 : e0 00 00 00 00 87 30 00  75
61a0 : 00 00 00 00 10 00 00 00  a2
61a8 : 00 00 20 00 00 00 00 00  b1
61b0 : 00 00 00 00 00 00 00 00  b1
61b8 : 00 00 00 00 00 00 00 00  b9
61c0 : 00 00 00 00 00 00 00 00  c1
61c8 : 00 00 00 00 00 00 00 00  c9
61d0 : 00 00 00 00 00 00 19 05  3f
61d8 : 80 00 20 40 00 00 3f ff  65
61e0 : e0 00 00 00 3f ff f0 00  78
61e8 : 00 00 40 00 10 00 00 00  fa
61f0 : 80 00 20 00 00 00 80 00  7b
61f8 : 00 00 00 00 00 00 00 00  f9
6200 : 00 00 00 00 00 00 00 00  01
6208 : 00 00 00 00 00 00 00 00  09
6210 : 00 00 00 00 00 00 00 00  11
6218 : 00 00 00 00 00 00 00 00  19
6220 : 00 00 00 00 00 00 00 00  21
6228 : 00 00 00 00 00 00 00 00  29
6230 : 00 00 00 00 00 00 00 00  31
6238 : 00 00 00 00 00 00 00 00  39
6240 : 00 00 00 00 00 00 00 00  41
6248 : 0b 05 01 00 00 00 00 00  16
6250 : 02 00 40 00 00 00 03 ff  6f
6258 : e0 00 00 00 00 ff f0 00  fc
6260 : 00 00 01 00 20 00 00 00  a3
6268 : 01 00 00 00 00 00 02 00  72
6270 : 00 00 00 00 02 00 40 00  92
6278 : 00 00 01 ff e0 00 00 00  c7
6280 : 00 ff f0 00 00 00 01 00  c0
6288 : 20 00 00 00 01 00 00 00  b9
6290 : 00 00 02 00 00 00 00 00  11
6298 : 02 00 20 00 00 00 01 ff  a7
62a0 : e0 00 00 00 00 ff f0 00  44
62a8 : 00 00 00 40 10 00 00 00  b2
62b0 : 00 00 20 00 00 00 00 00  b9
62b8 : 00 00 23 05 01 00 00 00  32
62c0 : 00 00 02 00 40 00 00 00  45
62c8 : 03 ff e0 00 00 00 00 ff  03
62d0 : f0 00 00 00 01 00 20 00  51
62d8 : 00 00 01 00 00 00 00 00  19
62e0 : 02 00 00 00 00 00 02 00  eb
62e8 : 00 00 00 00 01 ff e0 00  7c
62f0 : 00 00 00 ff f0 00 00 00  ff
62f8 : 00 40 10 00 00 00 00 00  1d
6300 : 20 00 00 00 00 00 00 00  21
6308 : 00 00 00 00 00 00 00 00  09
6310 : 00 00 00 00 00 00 00 00  11
6318 : 00 00 00 00 00 00 00 00  19
6320 : 00 00 00 00 00 00 00 00  21
6328 : 00 00 00 00 17 05 00 00  c2
6330 : 80 00 00 00 00 ff 80 00  b2
6338 : 00 00 01 ff c0 00 00 00  85
6340 : 01 00 40 00 00 00 01 00  56
6348 : 60 00 00 00 02 00 20 00  49
6350 : 00 00 02 00 30 00 00 00  d4
6358 : 03 00 10 00 00 00 03 00  6c
6360 : 20 00 c0 00 01 3f c0 00  be
6368 : c0 00 00 3f 80 00 00 00  18
6370 : 00 00 00 00 00 00 00 00  71
6378 : 00 00 00 00 00 00 00 00  79
6380 : 00 00 00 00 00 00 00 00  81
6388 : 00 00 00 00 00 00 00 00  89
6390 : 00 00 00 00 00 00 00 00  91
6398 : 00 00 00 00 00 00 15 05  f7
63a0 : 00 80 10 00 00 00 01 00  e9
63a8 : 20 00 00 00 03 ff ff 00  f9
63b0 : 00 00 01 ff ff 00 00 00  f1
63b8 : 01 00 30 00 00 00 01 00  ca
63c0 : 10 00 00 00 03 00 10 00  41
63c8 : 00 00 02 00 10 00 00 00  4a
63d0 : 03 80 70 00 00 00 01 ff  34
63d8 : e0 00 00 00 00 7f 80 00  b6
63e0 : 00 00 00 00 00 00 00 00  e1
63e8 : 00 00 00 00 00 00 00 00  e9
63f0 : 00 00 00 00 00 00 00 00  f1
63f8 : 00 00 00 00 00 00 00 00  f9
6400 : 00 00 00 00 00 00 00 00  01
6408 : 00 00 00 00 00 00 00 00  09
6410 : 15 05 00 7f 80 00 00 00  a0
6418 : 01 ff e0 00 00 00 03 80  5e
6420 : 70 00 00 00 02 00 10 00  f1
6428 : 00 00 02 00 10 00 00 00  aa
6430 : 02 00 10 00 00 00 03 00  43
6438 : 20 00 00 00 03 ff ff 00  89
6440 : 00 00 01 ff ff 00 00 00  81
6448 : 01 00 00 00 00 00 00 00  4a
6450 : 00 00 00 00 00 00 00 00  51
6458 : 00 00 00 00 00 00 00 00  59
6460 : 00 00 00 00 00 00 00 00  61
6468 : 00 00 00 00 00 00 00 00  69
6470 : 00 00 00 00 00 00 00 00  71
6478 : 00 00 00 00 00 00 00 00  79
6480 : 00 00 13 05 01 00 00 00  f6
6488 : 00 00 02 00 40 00 00 00  0d
6490 : 03 ff e0 00 00 00 00 ff  cb
6498 : f0 00 00 00 01 00 10 00  d9
64a0 : 00 00 03 00 20 00 00 00  63
64a8 : 03 80 00 00 00 00 01 00  f0
64b0 : 00 00 00 00 00 00 00 00  b1
64b8 : 00 00 00 00 00 00 00 00  b9
64c0 : 00 00 00 00 00 00 00 00  c1
64c8 : 00 00 00 00 00 00 00 00  c9
64d0 : 00 00 00 00 00 00 00 00  d1
64d8 : 00 00 00 00 00 00 00 00  d9
64e0 : 00 00 00 00 00 00 00 00  e1
64e8 : 00 00 00 00 00 00 00 00  e9
64f0 : 00 00 00 00 0f 05 00 fe  08
64f8 : 10 00 00 00 01 ff 20 00  99
6500 : 00 00 03 03 20 00 00 00  24
6508 : 02 02 20 00 00 00 03 04  28
6510 : 20 00 00 00 01 08 30 00  42
```

```
6518 : 00 00 01 10 10 00 00 00  5c
6520 : 01 30 30 00 00 00 01 3f  c8
6528 : e0 00 00 00 02 1f c0 00  25
6530 : 00 00 00 00 00 00 00 00  31
6538 : 00 00 00 00 00 00 00 00  39
6540 : 00 00 00 00 00 00 00 00  41
6548 : 00 00 00 00 00 00 00 00  49
6550 : 00 00 00 00 00 00 00 00  51
6558 : 00 00 00 00 00 00 00 00  59
6560 : 00 00 00 00 00 00 13 05  b7
6568 : 02 00 00 00 00 00 06 00  83
6570 : 40 00 00 00 1f ff e0 00  26
6578 : 00 00 ff ff f0 00 00 00  87
6580 : 02 00 10 00 00 00 02 00  8f
6588 : 20 00 00 00 00 00 00 00  a9
6590 : 00 00 00 00 00 00 00 00  91
6598 : 00 00 00 00 00 00 00 00  99
65a0 : 00 00 00 00 00 00 00 00  a1
65a8 : 00 00 00 00 00 00 00 00  a9
65b0 : 00 00 00 00 00 00 00 00  b1
65b8 : 00 00 00 00 00 00 00 00  b9
65c0 : 00 00 00 00 00 00 00 00  c1
65c8 : 00 00 00 00 00 00 00 00  c9
65d0 : 00 00 00 00 00 00 00 00  d1
65d8 : 0b 05 01 00 40 00 00 00  aa
65e0 : 03 ff c0 00 00 00 01 ff  17
65e8 : e0 00 00 00 00 80 20 00  4d
65f0 : 00 00 00 00 30 00 00 00  f4
65f8 : 00 00 10 00 00 00 00 00  fd
6600 : 10 00 00 00 01 00 20 00  a1
6608 : 00 00 03 ff e0 00 00 00  d7
6610 : 01 ff f0 00 00 00 00 80  4e
6618 : 10 00 00 00 00 00 20 00  a9
6620 : 00 00 00 00 00 00 00 00  21
6628 : 00 00 00 00 00 00 00 00  29
6630 : 00 00 00 00 00 00 00 00  31
6638 : 00 00 00 00 00 00 00 00  39
6640 : 00 00 00 00 00 00 00 00  41
6648 : 00 00 17 05 01 00 00 00  bf
6650 : 00 00 02 00 20 00 00 00  d3
6658 : 03 ff e0 00 00 00 01 ff  97
6660 : f0 00 00 00 01 00 30 00  21
6668 : 00 00 03 00 10 00 00 00  2a
6670 : 02 00 10 00 00 00 02 00  7f
6678 : 10 00 00 00 03 00 30 00  79
6680 : 00 00 01 ff e0 00 00 00  cf
6688 : 00 ff c0 00 00 00 00 00  b8
6690 : 00 00 00 00 00 00 00 00  91
6698 : 00 00 00 00 00 00 00 00  99
66a0 : 00 00 00 00 00 00 00 00  a1
66a8 : 00 00 00 00 00 00 00 00  a9
66b0 : 00 00 00 00 00 00 00 00  b1
66b8 : 00 00 00 00 15 05 01 00  36
66c0 : 00 00 00 00 02 00 20 00  61
66c8 : 00 00 03 ff e0 00 00 00  97
66d0 : 01 ff f0 00 00 00 01 00  11
66d8 : 30 00 00 00 03 00 10 00  79
66e0 : 00 00 02 00 30 00 00 00  64
66e8 : 03 ff e0 00 00 00 01 ff  27
66f0 : e0 00 00 00 01 00 20 00  61
66f8 : 00 00 03 00 30 00 00 00  bc
6700 : 02 00 10 00 00 00 03 00  13
6708 : 30 00 00 00 01 ff e0 00  cc
6710 : 00 00 00 7f 80 00 00 00  08
6718 : 00 00 00 00 00 00 00 00  19
6720 : 00 00 00 00 00 00 00 00  21
6728 : 00 00 00 00 00 00 1d 05  a7
6730 : 00 00 30 00 00 00 00 00  3d
6738 : 60 00 00 00 00 00 c0 00  9c
6740 : 00 00 01 01 80 00 00 00  a9
6748 : 02 03 00 00 00 00 03 ff  d8
6750 : e0 00 00 00 01 ff f0 00  04
6758 : 00 00 00 80 10 00 00 00  6a
6760 : 00 80 20 00 00 00 01 c0  2e
6768 : 00 00 00 00 00 80 00 00  6d
6770 : 00 00 00 00 00 00 00 00  71
6778 : 00 00 00 00 00 00 00 00  79
6780 : 00 00 00 00 00 00 00 00  81
6788 : 00 00 00 00 00 00 00 00  89
6790 : 00 00 00 00 00 00 00 00  91
6798 : 00 00 00 00 00 00 00 00  99
67a0 : 15 05 00 80 00 00 00 00  48
67a8 : 01 00 40 00 00 00 03 ff  c6
67b0 : e0 00 00 00 01 ff f1 00  68
67b8 : 00 00 01 00 11 00 00 00  0a
67c0 : 01 00 21 00 00 00 03 00  16
67c8 : 02 00 00 00 02 00 06 00  03
67d0 : 00 00 03 c0 1c 00 00 00  6b
67d8 : 01 ff f8 00 00 00 00 7f  16
67e0 : e0 00 00 00 00 00 00 00  c1
67e8 : 00 00 00 00 00 00 00 00  e9
67f0 : 00 00 00 00 00 00 00 00  f1
67f8 : 00 00 00 00 00 00 00 00  f9
6800 : 00 00 00 00 00 00 00 00  01
6808 : 00 00 00 00 00 00 00 00  09
6810 : 00 00 15 05 00 40 1c 00  69
6818 : 00 00 01 80 3e 00 00 00  4d
6820 : 03 04 43 00 00 00 02 04  06
6828 : 81 00 00 00 02 04 01 00  ee
6830 : 00 00 03 04 03 00 00 00  a2
6838 : 01 fb fe 00 00 00 00 70  d7
6840 : f8 00 00 00 00 00 00 00  39
6848 : 00 00 00 00 00 00 00 00  49
6850 : 00 00 00 00 00 00 00 00  51
6858 : 00 00 00 00 00 00 00 00  59
6860 : 00 00 00 00 00 00 00 00  61
6868 : 00 00 00 00 00 00 00 00  69
6870 : 00 00 00 00 00 00 00 00  71
6878 : 00 00 00 00 00 00 00 00  79
6880 : 00 00 00 00 0f 05 00 1f  d8
6888 : 80 00 60 60 00 1f 80 00  27
6890 : e0 70 20 00 00 51 00 30  9b
6898 : 20 00 00 43 00 10 00 00  a1
68a0 : 00 03 00 20 00 00 00 03  2c
68a8 : 80 20 00 00 00 21 80 40  c4
68b0 : 50 00 00 21 ff e0 40 00  2d
68b8 : 00 00 ff f0 00 00 00 00  d6
68c0 : 80 10 00 00 00 00 00 20  89
68c8 : 00 00 00 00 00 00 00 00  c9
68d0 : 00 00 00 00 00 00 00 00  d1
68d8 : 00 00 00 00 00 00 00 00  d9
68e0 : 00 00 00 00 00 00 00 00  e1
68e8 : 00 00 00 00 00 00 00 00  e9
68f0 : 00 00 00 00 00 00 16 05  53
68f8 : 00 00 80 00 00 00 00 ff  18
6900 : 80 00 00 00 01 ff c0 20  d4
6908 : 00 00 51 00 40 20 00 00  62
6910 : 41 00 60 00 00 00 02 00  72
6918 : 20 00 00 00 02 00 30 20  5a
6920 : 00 00 53 00 10 20 00 00  f7
6928 : 43 00 20 00 c0 00 01 3f  02
6930 : c0 00 c0 00 00 3f 80 00  1c
6938 : 00 00 00 00 00 00 00 00  39
6940 : 00 00 00 00 00 00 00 00  41
6948 : 00 00 00 00 00 00 00 00  49
6950 : 00 00 00 00 00 00 00 00  51
6958 : 00 00 00 00 00 00 00 00  59
6960 : 00 00 00 00 00 00 00 00  61
6968 : 15 05 01 00 40 00 00 00  44
6970 : 03 ff e0 00 00 00 21 ff  30
6978 : e0 50 00 00 20 80 30 40  c8
6980 : 00 00 00 00 10 00 00 00  82
6988 : 00 00 10 00 00 00 00 00  8d
6990 : 30 00 00 00 01 00 20 20  91
6998 : 00 00 53 ff e0 20 00 00  7c
69a0 : 41 ff f0 00 00 00 00 80  1e
69a8 : 10 00 00 00 00 00 20 00  39
69b0 : 00 00 00 00 00 00 00 00  b1
69b8 : 00 00 00 00 00 00 00 00  b9
69c0 : 00 00 00 00 00 00 00 00  c1
69c8 : 00 00 00 00 00 00 00 00  c9
69d0 : 00 00 00 00 00 00 00 00  d1
69d8 : 00 00 15 05 ff ff ff 00  be
69e0 : 00 00 81 ff 00 00 00 00  41
69e8 : c0 00 00 00 00 00 60 00  2a
69f0 : 00 00 00 00 60 00 00 00  f7
69f8 : 00 00 c0 00 00 00 00 00  29
6a00 : 00 00 00 00 00 00 00 00  01
6a08 : 00 00 00 00 00 00 00 00  09
6a10 : 00 00 00 00 00 00 00 00  11
6a18 : 00 00 00 00 00 00 00 00  19
6a20 : 00 00 00 00 00 00 00 00  21
6a28 : 00 00 00 00 00 00 00 00  29
6a30 : 00 00 00 00 00 00 00 00  31
6a38 : 00 00 00 00 00 00 00 00  39
6a40 : 00 00 00 00 00 00 00 00  41
6a48 : 00 00 00 00 0b 05 00 00  21
6a50 : 00 00 00 00 00 00 00 00  51
6a58 : 00 00 00 00 00 00 00 00  59
6a60 : 00 00 00 00 00 00 00 00  61
6a68 : 00 00 00 00 00 00 00 00  69
6a70 : 00 00 00 00 00 00 00 00  71
6a78 : 00 00 00 00 00 00 00 00  79
6a80 : 00 00 00 00 00 00 00 00  81
6a88 : 00 00 00 00 00 00 00 00  89
6a90 : 00 00 00 00 00 00 00 00  91
6a98 : 00 00 00 00 00 00 00 00  99
6aa0 : 00 00 00 00 00 00 00 00  a1
6aa8 : 00 00 00 00 00 00 00 00  a9
6ab0 : 00 00 00 00 00 00 00 00  b1
6ab8 : 00 00 00 00 00 00 00 00  b9
6ac0 : 20 00 00 00 00 00 00 00  e1
```

Listing 2. (Schluß)

```
Name : english/h20    .a  4000 6ac1
-----------------------------------
4000 : 00 00 00 00 00 00 00 00  01
4008 : 00 00 00 00 00 00 00 00  09
4010 : 00 00 00 00 00 00 00 00  11
4018 : 00 00 00 00 00 00 00 00  19
4020 : 00 00 00 00 00 00 00 00  21
4028 : 00 00 00 00 00 00 00 00  29
4030 : 00 00 00 00 00 00 00 00  31
4038 : 00 00 00 00 00 00 00 00  39
4040 : 00 00 00 00 00 00 00 00  41
4048 : 00 00 00 00 00 00 00 00  49
4050 : 00 00 00 00 00 00 00 00  51
4058 : 00 00 00 00 00 00 00 00  59
4060 : 00 00 00 00 00 00 00 00  61
4068 : 00 00 00 00 00 00 00 00  69
4070 : 15 05 1f 00 00 00 00 20  10
4078 : 7f e0 00 00 00 70 ff fc  e5
4080 : 00 00 00 70 7f e0 00 00  8d
4088 : 00 20 1f 80 00 00 00 00  70
4090 : 00 00 00 00 00 00 00 00  91
4098 : 00 00 00 00 00 00 00 00  99
40a0 : 00 00 00 00 00 00 00 00  a1
40a8 : 00 00 00 00 00 00 00 00  a9
40b0 : 00 00 00 00 00 00 00 00  b1
40b8 : 00 00 00 00 00 00 00 00  b9
40c0 : 00 00 00 00 00 00 00 00  c1
40c8 : 00 00 00 00 00 00 00 00  c9
40d0 : 00 00 00 00 00 00 00 00  d1
40d8 : 00 00 00 00 00 00 00 00  d9
```

Listing 3. »ENGLISH/H20.A« gefällt durch seine geschwungene Schriftformen

```
40e0 : 00 00 09 05 40 00 00 00  c7
40e8 : 00 00 f0 00 00 00 00 00  25
40f0 : e0 00 00 00 00 00 40 00  d2
40f8 : 00 00 00 00 00 00 00 00  f9
4100 : 00 00 40 00 00 00 00 00  11
4108 : f0 00 00 00 00 00 e0 00  7c
4110 : 00 00 00 00 40 00 00 00  15
4118 : 00 00 00 00 00 00 00 00  19
4120 : 00 00 00 00 00 00 00 00  21
4128 : 00 00 00 00 00 00 00 00  29
4130 : 00 00 00 00 00 00 00 00  31
4138 : 00 00 00 00 00 00 00 00  39
4140 : 00 00 00 00 00 00 00 00  41
4148 : 00 00 00 00 00 00 00 00  49
4150 : 00 00 00 00 11 05 00 00  8a
4158 : 00 00 00 00 00 00 00 00  59
4160 : 00 00 00 00 00 00 00 00  61
4168 : 00 00 00 00 00 00 00 00  69
4170 : 00 00 00 00 00 00 00 00  71
4178 : 00 00 00 00 00 00 00 00  79
4180 : 00 00 00 00 00 00 00 00  81
4188 : 00 00 00 00 00 00 00 00  89
4190 : 00 00 00 00 00 00 00 00  91
4198 : 00 00 00 00 00 00 00 00  99
41a0 : 00 00 00 00 00 00 00 00  a1
41a8 : 00 00 00 00 00 00 00 00  a9
41b0 : 00 00 00 00 00 00 00 00  b1
41b8 : 00 00 00 00 00 00 00 00  b9
41c0 : 00 00 00 00 00 00 00 00  c1
41c8 : 00 00 00 00 00 00 00 00  c9
41d0 : 00 00 00 00 00 00 00 00  d1
41d8 : 00 00 00 00 00 00 00 00  d9
41e0 : 00 00 00 00 00 00 00 00  e1
41e8 : 00 00 00 00 00 00 00 00  e9
41f0 : 00 00 00 00 00 00 00 00  f1
41f8 : 00 00 00 00 00 00 00 00  f9
4200 : 00 00 00 00 00 00 00 00  01
4208 : 00 00 00 00 00 00 00 00  09
4210 : 00 00 00 00 00 00 00 00  11
4218 : 00 00 00 00 00 00 00 00  19
4220 : 00 00 00 00 00 00 00 00  21
4228 : 00 00 00 00 00 00 00 00  29
4230 : 00 00 00 00 00 00 00 00  31
4238 : 00 00 00 00 70 00 00 00  40
4240 : 38 01 c0 00 00 00 64 03  c1
4248 : 80 00 00 00 78 06 00 00  80
4250 : 00 00 30 0c 00 00 00 00  de
4258 : 00 18 00 00 00 00 00 30  c5
4260 : 00 00 00 00 00 60 00 00  64
4268 : 00 00 00 c0 00 00 00 00  81
4270 : 01 80 00 00 00 00 03 01  c0
4278 : c0 00 00 00 06 03 20 00  31
4280 : 00 00 1c 03 c0 00 00 00  f4
4288 : 38 01 80 00 00 00 e0 00  e5
4290 : 00 00 00 00 00 00 00 00  91
4298 : 00 00 00 00 00 00 00 00  99
42a0 : 00 00 00 00 00 00 00 00  a1
42a8 : 00 00 1d 05 00 0f c0 00  0c
42b0 : 00 00 00 1f e0 00 00 00  a2
42b8 : 00 30 70 00 00 00 0f 20  69
42c0 : 30 00 00 00 1f c0 30 00  a9
42c8 : 00 00 11 e0 30 00 00 00  2c
42d0 : 11 f8 20 00 00 00 11 7e  a7
42d8 : 40 00 00 00 0e 1f 80 00  f4
42e0 : 00 00 00 0f e0 00 00 00  d0
42e8 : 00 03 f0 00 00 00 00 04  ae
42f0 : f0 00 00 00 00 08 30 00  e2
42f8 : 00 00 00 08 10 00 00 00  fb
4300 : 00 04 20 00 00 00 00 00  0b
4308 : 00 00 00 00 00 00 00 00  09
4310 : 00 00 00 00 00 00 00 00  11
4318 : 00 00 00 00 1d 05 00 00  12
4320 : 00 00 00 00 00 00 00 00  21
4328 : 00 00 00 00 00 00 00 00  29
4330 : 00 00 00 00 00 00 00 00  31
```

```
4338 : 00 00 00 00 00 00 00 00  39
4340 : 00 00 00 00 00 00 00 00  41
4348 : 00 00 00 00 00 00 00 00  49
4350 : 00 00 00 00 00 00 00 00  51
4358 : 00 00 00 00 00 00 00 00  59
4360 : 00 00 00 00 00 00 00 00  61
4368 : 00 00 00 00 00 00 00 00  69
4370 : 00 00 00 00 00 00 00 00  71
4378 : 00 00 00 00 00 00 00 00  79
4380 : 00 00 00 00 00 00 00 00  81
4388 : 00 00 00 00 00 00 00 00  89
4390 : 00 f0 00 00 00 00 03 fc  0f
4398 : 00 00 00 00 1c 03 80 00  74
43a0 : 00 00 60 00 60 00 00 00  bf
43a8 : 80 00 10 00 00 00 00 00  2d
43b0 : 00 00 00 00 00 00 00 00  b1
43b8 : 00 00 00 00 00 00 00 00  b9
43c0 : 00 00 00 00 00 00 00 00  c1
43c8 : 00 00 00 00 00 00 00 00  c9
43d0 : 00 00 00 00 00 00 00 00  d1
43d8 : 00 00 00 00 00 00 00 00  d9
43e0 : 00 00 00 00 00 00 00 00  e1
43e8 : 00 00 00 00 00 00 00 00  e9
43f0 : 00 00 00 00 00 00 00 00  f1
43f8 : 00 00 00 00 00 00 00 00  f9
4400 : 09 05 80 00 10 00 00 00  ad
4408 : 60 00 60 00 00 00 1c 03  f7
4410 : 80 00 00 00 03 fc 00 00  a9
4418 : 00 00 00 f0 00 00 00 00  37
4420 : 00 00 00 00 00 00 00 00  21
4428 : 00 00 00 00 00 00 00 00  29
4430 : 00 00 00 00 00 00 00 00  31
4438 : 00 00 00 00 00 00 00 00  39
4440 : 00 00 00 00 00 00 00 00  41
4448 : 00 00 00 00 00 00 00 00  49
4450 : 00 00 00 00 00 00 00 00  51
4458 : 00 00 00 00 00 00 00 00  59
4460 : 00 00 00 00 00 00 00 00  61
4468 : 00 00 00 00 00 00 00 00  69
4470 : 00 00 09 05 00 7c 00 00  37
4478 : 00 00 02 10 80 00 00 00  03
4480 : 03 11 80 00 00 00 00 92  51
4488 : 00 00 00 00 00 54 00 00  2b
4490 : 00 00 00 38 00 00 00 00  98
4498 : 00 54 00 00 00 00 00 92  e8
44a0 : 00 00 00 00 03 11 80 00  5b
44a8 : 00 00 02 10 80 00 00 00  33
44b0 : 00 7c 00 00 00 00 00 00  ef
44b8 : 00 00 00 00 00 00 00 00  b9
44c0 : 00 00 00 00 00 00 00 00  c1
44c8 : 00 00 00 00 00 00 00 00  c9
44d0 : 00 00 00 00 00 00 00 00  d1
44d8 : 00 00 00 00 00 00 00 00  d9
44e0 : 00 00 00 00 15 05 00 10  7a
44e8 : 00 00 00 00 00 10 00 00  69
44f0 : 00 00 00 10 00 00 00 00  f3
44f8 : 01 fe 00 00 00 00 00 ff  78
4500 : 00 00 00 00 00 10 00 00  81
4508 : 00 00 00 10 00 00 00 00  0b
4510 : 00 10 00 00 00 00 00 10  39
4518 : 00 00 00 00 00 00 00 00  19
4520 : 00 00 00 00 00 00 00 00  21
4528 : 00 00 00 00 00 00 00 00  29
4530 : 00 00 00 00 00 00 00 00  31
4538 : 00 00 00 00 00 00 00 00  39
4540 : 00 00 00 00 00 00 00 00  41
4548 : 00 00 00 00 00 00 00 00  49
4550 : 00 00 00 00 00 00 11 05  9f
4558 : 00 00 24 00 00 00 00 00  62
4560 : 78 00 00 00 00 00 70 00  9a
4568 : 00 00 00 00 20 00 00 00  6b
4570 : 00 00 00 00 00 00 00 00  71
4578 : 00 00 00 00 00 00 00 00  79
4580 : 00 00 00 00 00 00 00 00  81
4588 : 00 00 00 00 00 00 00 00  89
```

```
4590 : 00 00 00 00 00 00 00 00  91
4598 : 00 00 00 00 00 00 00 00  99
45a0 : 00 00 00 00 00 00 00 00  a1
45a8 : 00 00 00 00 00 00 00 00  a9
45b0 : 00 00 00 00 00 00 00 00  b1
45b8 : 00 00 00 00 00 00 00 00  b9
45c0 : 00 00 00 00 00 00 00 00  c1
45c8 : 07 05 00 08 00 00 00 00  53
45d0 : 00 04 00 00 00 00 00 06  df
45d8 : 00 00 00 00 00 06 00 00  09
45e0 : 00 00 00 06 00 00 00 00  a1
45e8 : 00 04 00 00 00 00 00 0c  03
45f0 : 00 00 00 00 00 0c 00 00  51
45f8 : 00 00 00 0c 00 00 00 00  7a
4600 : 00 04 00 00 00 00 00 02  07
4608 : 00 00 00 00 00 00 00 00  09
4610 : 00 00 00 00 00 00 00 00  11
4618 : 00 00 00 00 00 00 00 00  19
4620 : 00 00 00 00 00 00 00 00  21
4628 : 00 00 00 00 00 00 00 00  29
4630 : 00 00 00 00 00 00 00 00  31
4638 : 00 00 15 05 00 00 20 00  9f
4640 : 00 00 00 00 70 00 00 00  48
4648 : 00 00 70 00 00 00 00 00  65
4650 : 20 00 00 00 00 00 00 00  71
4658 : 00 00 00 00 00 00 00 00  59
4660 : 00 00 00 00 00 00 00 00  61
4668 : 00 00 00 00 00 00 00 00  69
4670 : 00 00 00 00 00 00 00 00  71
4678 : 00 00 00 00 00 00 00 00  79
4680 : 00 00 00 00 00 00 00 00  81
4688 : 00 00 00 00 00 00 00 00  89
4690 : 00 00 00 00 00 00 00 00  91
4698 : 00 00 00 00 00 00 00 00  99
46a0 : 00 00 00 00 00 00 00 00  a1
46a8 : 00 00 00 00 07 05 00 00  41
46b0 : 70 00 00 00 00 01 c0 00  2c
46b8 : 00 00 00 03 80 00 00 00  21
46c0 : 00 06 00 00 00 00 00 0c  dc
46c8 : 00 00 00 00 00 18 00 00  89
46d0 : 00 00 00 70 00 00 00 00  df
46d8 : 00 e0 00 00 00 00 03 80  56
46e0 : 00 00 00 00 00 00 00 00  e1
46e8 : 00 00 00 00 00 00 00 00  e9
46f0 : 00 00 00 00 00 00 00 00  f1
46f8 : 00 00 00 00 00 00 00 00  f9
4700 : 00 00 00 00 00 00 00 00  01
4708 : 00 00 00 00 00 00 00 00  09
4710 : 00 00 00 00 00 00 00 00  11
4718 : 00 00 00 00 00 00 11 05  67
4720 : 00 fe 00 00 00 00 03 ff  ac
4728 : 80 00 00 00 0f ff c0 00  9c
4730 : 00 00 1f 01 e0 00 00 00  26
4738 : 38 00 20 00 00 00 60 00  fa
4740 : 10 00 00 00 40 00 10 00  95
4748 : 00 00 80 00 10 00 00 00  6a
4750 : 80 00 20 00 00 00 80 00  db
4758 : 60 00 00 00 40 01 c0 00  c8
4760 : 00 00 78 0f 80 00 00 00  68
4768 : 3f ff 00 00 00 00 1f fc  1d
4770 : 00 00 00 00 07 f0 00 00  68
4778 : 00 00 00 00 00 00 00 00  79
4780 : 00 00 00 00 00 00 00 00  81
4788 : 00 00 00 00 00 00 00 00  89
4790 : 1d 05 08 00 00 00 00 00  32
4798 : 10 00 08 00 00 00 20 00  2b
47a0 : 10 00 00 00 7f ff f0 00  6c
47a8 : 00 00 ff ff e0 00 00 00  b6
47b0 : 00 00 20 00 00 00 00 00  b9
47b8 : 00 00 00 00 00 00 00 00  b9
47c0 : 00 00 00 00 00 00 00 00  c1
47c8 : 00 00 00 00 00 00 00 00  c9
47d0 : 00 00 00 00 00 00 00 00  d1
47d8 : 00 00 00 00 00 00 00 00  d9
47e0 : 00 00 00 00 00 00 00 00  e1
```

```
47e8 : 00 00 00 00 00 00 00 00  e9
47f0 : 00 00 00 00 00 00 00 00  f1
47f8 : 00 00 00 00 00 00 00 00  f9
4800 : 00 00 0b 05 00 00 30 00  25
4808 : 00 00 18 01 e0 00 00 00  3d
4810 : 20 07 e0 00 00 00 40 0e  09
4818 : 60 00 00 00 40 1c 60 00  df
4820 : 00 00 c0 18 60 00 00 00  5a
4828 : c0 30 40 00 00 00 c0 30  74
4830 : c0 00 00 00 e0 60 c0 00  05
4838 : 00 00 f0 e0 c0 00 00 00  9d
4840 : 7f c0 c0 00 00 00 7f c0  cf
4848 : 40 00 00 00 3f 80 60 00  02
4850 : 00 00 0e 00 20 00 00 00  d6
4858 : 00 00 10 00 00 00 00 00  5d
4860 : 00 00 00 00 00 00 00 00  61
4868 : 00 00 00 00 00 00 00 00  69
4870 : 00 00 00 00 1d 05 00 01  6c
4878 : 00 00 00 00 0e 03 c0 00  74
4880 : 00 00 32 00 60 00 00 00  13
4888 : 60 00 30 00 00 00 c0 40  78
4890 : 30 00 00 00 c0 40 30 00  8f
4898 : 00 00 c0 40 30 00 00 00  d4
48a0 : c0 a0 70 00 00 00 61 b8  c3
48a8 : f0 00 00 00 7f 1f e0 00  0d
48b0 : 00 00 3e 0f c0 00 00 00  2e
48b8 : 0c 03 80 00 00 00 00 00  66
48c0 : 00 00 00 00 00 00 00 00  c1
48c8 : 00 00 00 00 00 00 00 00  c9
48d0 : 00 00 00 00 00 00 00 00  d1
48d8 : 00 00 00 00 00 00 00 00  d9
48e0 : 00 00 00 00 00 00 15 05  3f
48e8 : 00 08 00 00 00 00 00 18  1d
48f0 : 00 00 00 00 00 38 00 00  b2
48f8 : 00 00 00 58 00 00 00 00  04
4900 : 00 98 00 00 00 00 01 18  81
4908 : 00 00 00 00 02 18 00 00  e9
4910 : 00 00 04 18 00 00 00 00  15
4918 : 08 18 00 00 00 00 10 18  9d
4920 : 10 00 00 00 20 18 20 00  74
4928 : 00 00 7f ff e0 00 00 00  16
4930 : ff ff c0 00 00 00 00 18  8f
4938 : 40 00 00 00 00 18 00 00  39
4940 : 00 00 00 10 00 00 00 00  43
4948 : 00 00 00 00 00 00 00 00  49
4950 : 00 00 00 00 00 00 00 00  51
4958 : 1f 05 01 c0 c0 00 00 00  5e
4960 : 03 81 e0 00 00 00 0f 00  98
4968 : e0 00 00 00 33 00 30 00  3c
4970 : 00 00 c3 00 10 00 00 00  62
4978 : c6 00 10 00 00 00 c6 00  5e
4980 : 10 00 00 00 c6 00 10 00  3d
4988 : 00 00 63 00 20 00 00 00  63
4990 : 63 80 60 00 00 00 61 e1  95
4998 : c0 00 00 00 40 ff 80 00  5e
49a0 : 00 00 80 1e 00 00 00 00  84
49a8 : 00 00 00 00 00 00 00 00  a9
49b0 : 00 00 00 00 00 00 00 00  b1
49b8 : 00 00 00 00 00 00 00 00  b9
49c0 : 00 00 00 00 00 00 00 00  c1
49c8 : 00 00 19 05 00 7c 00 00  93
49d0 : 00 00 01 ff 00 00 00 00  11
49d8 : 07 ff c0 00 00 00 0f 11  6d
49e0 : e0 00 00 00 1c 20 60 00  05
49e8 : 00 00 30 40 30 00 00 00  00
49f0 : 60 40 10 00 00 00 40 40  f6
49f8 : 10 00 00 00 80 60 30 00  d4
4a00 : 00 00 80 20 20 00 00 00  27
4a08 : 00 38 e0 00 00 00 00 1f  9b
4a10 : c0 00 00 00 00 07 00 00  09
4a18 : 00 00 00 00 00 00 00 00  19
4a20 : 00 00 00 00 00 00 00 00  21
4a28 : 00 00 00 00 00 00 00 00  29
4a30 : 00 00 00 00 00 00 00 00  31
4a38 : 00 00 00 00 19 05 20 00  73
4a40 : 00 00 00 00 40 00 00 00  45
4a48 : 00 00 c0 00 00 00 00 00  79
4a50 : c0 01 e0 00 00 00 c0 1f  0a
4a58 : f0 00 00 00 c0 ff e0 00  d8
4a60 : 00 00 c3 f0 00 00 00 00  6f
4a68 : c7 80 00 00 00 00 ce 00  ab
4a70 : 00 00 00 00 d8 00 00 00  fe
4a78 : 00 00 f0 00 00 00 00 00  b5
4a80 : e0 00 00 00 00 00 c0 00  64
4a88 : 00 00 00 00 80 00 00 00  91
4a90 : 00 00 00 00 00 00 00 00  91
4a98 : 00 00 00 00 00 00 00 00  99
4aa0 : 00 00 00 00 00 00 00 00  a1
4aa8 : 00 00 00 00 00 00 1b 05  1f
4ab0 : 1c 0f 00 00 00 00 2f 13  37
4ab8 : 80 00 00 00 47 a1 c0 00  bd
4ac0 : 00 00 c3 c0 e0 00 00 00  d7
4ac8 : c1 e0 e0 00 00 00 e1 e0  7b
4ad0 : 70 00 00 00 e0 f0 70 00  98
4ad8 : 00 00 70 f0 30 00 00 00  16
4ae0 : 70 78 30 00 00 00 38 bc  f3
4ae8 : 20 00 00 00 1d 1e 40 00  cc
4af0 : 00 00 0e 07 80 00 00 00  5d
4af8 : 00 00 00 00 00 00 00 00  f9
4b00 : 00 00 00 00 00 00 00 00  01
4b08 : 00 00 00 00 00 00 00 00  09
4b10 : 00 00 00 00 00 00 00 00  11
4b18 : 00 00 00 00 00 00 00 00  19
4b20 : 17 05 0e 00 00 00 00 00  3e
4b28 : 3f 80 00 00 00 00 71 c0  ef
4b30 : 00 00 00 00 40 40 10 00  77
4b38 : 00 00 c0 60 10 00 00 00  76
4b40 : 80 20 20 00 00 00 80 20  1b
4b48 : 60 00 00 00 c0 20 c0 00  b9
4b50 : 00 00 60 43 80 00 00 00  d9
4b58 : 78 8f 00 00 00 00 3f fe  93
4b60 : 00 00 00 00 0f f8 00 00  19
4b68 : 00 00 03 e0 00 00 00 00  45
4b70 : 00 00 00 00 00 00 00 00  71
4b78 : 00 00 00 00 00 00 00 00  79
4b80 : 00 00 00 00 00 00 00 00  81
4b88 : 00 00 00 00 00 00 00 00  89
4b90 : 00 00 19 05 00 18 60 00  ba
4b98 : 00 00 00 3c f0 00 00 00  2f
4ba0 : 00 3c f0 00 00 00 00 18  2b
4ba8 : 60 00 00 00 00 00 00 00  09
4bb0 : 00 00 00 00 00 00 00 00  b1
4bb8 : 00 00 00 00 00 00 00 00  b9
4bc0 : 00 00 00 00 00 00 00 00  c1
4bc8 : 00 00 00 00 00 00 00 00  c9
4bd0 : 00 00 00 00 00 00 00 00  d1
4bd8 : 00 00 00 00 00 00 00 00  d9
4be0 : 00 00 00 00 00 00 00 00  e1
4be8 : 00 00 00 00 00 00 00 00  e9
4bf0 : 00 00 00 00 00 00 00 00  f1
4bf8 : 00 00 00 00 00 00 00 00  f9
4c00 : 00 00 00 00 07 05 00 18  c9
4c08 : 64 00 00 00 00 3c e8 00  f2
4c10 : 00 00 00 3c f0 00 00 00  a7
4c18 : 00 18 60 00 00 00 00 00  3d
4c20 : 00 00 00 00 00 00 00 00  21
4c28 : 00 00 00 00 00 00 00 00  29
4c30 : 00 00 00 00 00 00 00 00  31
4c38 : 00 00 00 00 00 00 00 00  39
4c40 : 00 00 00 00 00 00 00 00  41
4c48 : 00 00 00 00 00 00 00 00  49
4c50 : 00 00 00 00 00 00 00 00  51
4c58 : 00 00 00 00 00 00 00 00  59
4c60 : 00 00 00 00 00 00 00 00  61
4c68 : 00 00 00 00 00 00 00 00  69
4c70 : 00 00 00 00 00 00 07 05  97
4c78 : 00 00 00 00 00 00 00 00  79
4c80 : 00 00 00 00 00 00 00 00  81
4c88 : 00 00 00 00 00 00 00 00  89
4c90 : 00 00 00 00 00 00 00 00  91
4c98 : 00 00 00 00 00 00 00 00  99
4ca0 : 00 00 00 00 00 00 00 00  a1
4ca8 : 00 00 00 00 00 00 00 00  a9
4cb0 : 00 00 00 00 00 00 00 00  b1
4cb8 : 00 00 00 00 00 00 00 00  b9
4cc0 : 00 00 00 00 00 00 00 00  c1
4cc8 : 00 00 00 00 00 00 00 00  c9
4cd0 : 00 00 00 00 00 00 00 00  d1
4cd8 : 00 00 00 00 00 00 00 00  d9
4ce0 : 00 00 00 00 00 00 00 00  e1
4ce8 : 00 00 00 12 00 00 00 00  2b
4cf0 : 00 09 00 00 00 00 00 09  87
4cf8 : 00 00 00 00 00 09 00 00  41
4d00 : 00 00 00 1b 00 00 00 00  64
4d08 : 00 1b 00 00 00 00 00 12  ba
4d10 : 00 00 00 00 00 09 00 00  59
4d18 : 00 00 00 00 00 00 00 00  19
4d20 : 00 00 00 00 00 00 00 00  21
4d28 : 00 00 00 00 00 00 00 00  29
4d30 : 00 00 00 00 00 00 00 00  31
4d38 : 00 00 00 00 00 00 00 00  39
4d40 : 00 00 00 00 00 00 00 00  41
4d48 : 00 00 00 00 00 00 00 00  49
4d50 : 00 00 00 00 00 00 00 00  51
4d58 : 00 00 0f 05 00 00 00 00  bd
4d60 : 00 00 00 00 00 00 00 00  61
4d68 : 00 00 00 00 00 00 00 00  69
4d70 : 00 00 00 00 00 00 00 00  71
4d78 : 00 00 00 00 00 00 00 00  79
4d80 : 00 00 00 00 00 00 00 00  81
4d88 : 00 00 00 00 00 00 00 00  89
4d90 : 00 00 00 00 00 00 00 00  91
4d98 : 00 00 00 00 00 00 00 00  99
4da0 : 00 00 00 00 00 00 00 00  a1
4da8 : 00 00 00 00 00 00 00 00  a9
4db0 : 00 00 00 00 00 00 00 00  b1
4db8 : 00 00 00 00 00 00 00 00  b9
4dc0 : 00 00 00 00 00 00 00 00  c1
4dc8 : 00 00 00 00 00 00 00 00  c9
4dd0 : 00 00 00 00 00 00 00 00  d1
4dd8 : 00 00 00 00 00 00 00 00  d9
4de0 : 00 00 00 00 00 00 00 00  e1
4de8 : 00 00 00 00 00 00 00 00  e9
4df0 : 00 00 00 00 00 00 00 00  f1
4df8 : 00 00 00 00 00 00 00 00  f9
4e00 : 00 00 00 00 00 00 00 00  01
4e08 : 00 00 00 00 00 00 00 00  09
4e10 : 00 00 00 00 00 00 00 00  11
4e18 : 00 00 00 00 00 00 00 00  19
4e20 : 00 00 00 00 00 00 00 00  21
4e28 : 00 00 00 00 00 00 00 00  29
4e30 : 00 00 00 00 00 00 00 00  31
4e38 : 00 00 00 00 00 00 00 00  39
4e40 : 00 00 00 00 00 00 00 00  41
4e48 : 00 00 00 00 00 00 00 00  49
4e50 : 00 00 00 00 00 00 00 00  51
4e58 : 00 00 00 00 00 00 00 00  59
4e60 : 00 00 00 00 00 00 00 00  61
4e68 : 00 00 00 00 00 00 00 00  69
4e70 : 00 00 00 00 00 00 00 00  71
4e78 : 00 00 00 00 00 00 00 00  79
4e80 : 00 00 00 00 00 00 00 00  81
4e88 : 00 00 00 00 00 00 00 00  89
4e90 : 00 00 00 00 00 00 00 00  91
4e98 : 00 00 00 00 00 00 00 00  99
4ea0 : 00 00 00 00 00 00 00 00  a1
4ea8 : 00 00 00 00 00 00 00 00  a9
4eb0 : 00 00 08 00 20 00 00 00  b5
4eb8 : 1f ff e0 00 00 00 3f ff  0c
4ec0 : f0 00 00 00 00 00 30 00  71
4ec8 : 00 00 7f ff f0 00 00 00  b7
4ed0 : c0 00 20 00 20 00 c0 00  9e
```

Listing 3. (Fortsetzung)

```
4ed8 : 00 00 60 00 c0 00 00 00   fd
4ee0 : 60 00 c0 00 00 00 60 00   f2
4ee8 : e0 00 00 00 40 00 60 00   4e
4ef0 : 40 00 00 00 7f ff e0 00   ac
4ef8 : 00 00 7f ff f0 00 00 00   e7
4f00 : 60 00 20 00 00 00 c0 00   6c
4f08 : 00 00 00 00 00 00 00 00   09
4f10 : 00 00 00 00 00 00 00 00   11
4f18 : 00 00 00 00 00 00 00 00   19
4f20 : 00 03 1d 03 20 02 00 00   5c
4f28 : 00 00 70 07 00 00 00 00   25
4f30 : f8 0b 00 00 00 80 3e 11   cd
4f38 : 00 00 60 c0 0f 80 00 00   5d
4f40 : 00 c0 00 00 20 00 00 40   23
4f48 : 20 00 20 00 00 50 7f ff   f1
4f50 : 20 00 00 10 ff fe 20 00   eb
4f58 : 00 10 20 00 00 00 80 10   8b
4f60 : 41 00 00 00 80 10 e2 00   b6
4f68 : 20 00 80 10 74 00 20 00   72
4f70 : c0 10 38 20 60 00 c0 80   55
4f78 : 10 3f 60 00 c0 80 00 3f   ce
4f80 : 40 00 40 80 00 3f 00 00   da
4f88 : 00 00 00 00 00 00 00 00   89
4f90 : 00 00 00 00 21 05 03 fc   d1
4f98 : 00 00 00 00 0f ff 00 00   89
4fa0 : 00 00 10 0f 80 00 00 00   8e
4fa8 : 20 03 c0 00 00 00 40 01   7d
4fb0 : e0 00 00 00 7f fe e0 00   04
4fb8 : 00 00 ff fc 70 00 00 00   5f
4fc0 : 00 00 30 00 00 00 00 00   cd
4fc8 : 30 00 00 00 3f ff f0 00   b0
4fd0 : 00 00 40 00 10 00 00 00   e2
4fd8 : e0 00 30 00 00 00 e0 00   48
4fe0 : 20 00 00 00 70 00 60 00   89
4fe8 : 00 00 70 00 40 00 00 00   09
4ff0 : 60 00 80 00 00 00 40 01   74
4ff8 : 00 00 00 00 00 00 00 00   f9
5000 : 00 00 00 00 00 00 1f 05   87
5008 : 20 00 10 00 00 00 40 00   2e
5010 : 20 00 00 00 c0 00 60 00   be
5018 : 00 00 cf ff 60 00 00 00   12
5020 : df fe 60 00 00 00 c0 00   9a
5028 : 60 00 00 00 c0 00 20 00   15
5030 : 00 00 ff ff f0 00 00 00   3f
5038 : e0 00 30 00 00 00 e0 00   a8
5040 : 30 00 00 00 e0 00 30 00   3f
5048 : 00 00 60 00 30 00 00 00   64
5050 : 60 00 20 00 00 00 30 00   79
5058 : c0 00 00 00 3f ff 80 00   0e
5060 : 00 00 1f ff 00 00 00 00   28
5068 : 0f fe 00 00 00 00 00 00   f7
5070 : 00 00 00 00 00 00 00 00   71
5078 : 21 05 03 fc 00 00 00 00   7c
5080 : 0f ff 00 00 00 00 10 0f   ed
5088 : 80 00 00 00 20 03 c0 00   26
5090 : 00 00 40 01 e0 00 00 00   cf
5098 : 7f fe e0 00 00 00 ff fc   c8
50a0 : 70 00 00 00 00 00 70 00   d2
50a8 : 00 00 00 00 30 00 00 00   ac
50b0 : 3f ff f0 00 00 00 40 40   ad
50b8 : 10 00 00 00 e0 80 30 00   9b
50c0 : 00 00 e0 c0 20 00 00 00   13
50c8 : 70 c0 60 00 00 00 70 80   73
50d0 : 40 00 00 00 60 00 80 00   19
50d8 : 00 00 40 01 00 00 00 00   09
50e0 : 00 00 00 00 00 00 00 00   e1
50e8 : 00 00 1f 05 00 30 01 00   d6
50f0 : 00 00 00 60 01 00 00 00   0d
50f8 : 04 60 02 00 00 00 08 60   8e
5100 : 06 00 00 00 1f ff fc 00   ec
5108 : 00 00 3f ff f8 00 00 00   68
5110 : 00 60 10 00 00 00 7f ff   42
5118 : c0 00 00 00 e0 60 00 00   ea
5120 : 00 00 f0 60 00 00 00 00   69
5128 : 78 60 00 00 00 00 30 40   12
5130 : 00 00 00 00 20 00 00 00   33
5138 : 00 00 40 00 00 00 00 00   49
5140 : 80 00 00 00 00 00 00 00   c1
5148 : 00 00 00 00 00 00 00 00   49
5150 : 00 00 00 00 00 00 00 00   51
5158 : 00 00 00 00 1d 05 03 fc   58
5160 : 00 00 00 00 0f ff 00 00   51
5168 : 00 00 10 1f 80 00 00 00   58
5170 : 20 07 c0 00 00 00 40 01   47
5178 : e0 00 00 00 7f fe e0 00   cc
5180 : 00 00 ff fc 70 00 00 00   27
5188 : 00 00 70 00 00 00 00 00   a5
5190 : 30 00 00 00 ff ff f0 00   84
5198 : 00 00 20 00 10 00 80 00   a4
51a0 : 41 00 30 00 80 00 e2 00   81
51a8 : 30 00 80 00 74 00 30 00   01
51b0 : 80 00 38 40 60 00 80 00   4f
51b8 : 10 40 60 00 00 80 00 7f   03
51c0 : 40 00 00 80 00 3f 00 00   0a
51c8 : 00 00 00 00 00 00 23 05   5f
51d0 : 20 00 01 00 00 00 40 00   32
51d8 : 01 00 00 00 c0 00 02 00   ee
51e0 : 00 00 c4 00 06 00 00 00   72
51e8 : cf ff fc 00 00 00 df ff   75
51f0 : f8 00 00 00 e0 02 10 00   47
51f8 : 00 00 ff ff c0 00 00 00   04
5200 : 91 12 00 00 00 00 22 12   47
5208 : 00 00 00 00 44 12 01 00   e1
5210 : 00 00 8c 12 01 00 00 00   86
5218 : 0e 12 02 00 00 00 07 12   f0
5220 : 06 00 00 00 03 ff fc 00   4b
5228 : 00 00 01 ff f8 00 00 00   f8
5230 : 00 80 10 00 00 00 00 00   75
5238 : 00 00 00 00 00 00 00 00   39
5240 : 21 05 00 00 10 00 00 00   e5
5248 : 20 00 20 00 00 00 40 00   72
5250 : 40 00 00 00 c0 00 e0 00   20
5258 : 00 00 c0 00 70 00 00 00   90
5260 : 61 00 20 00 00 00 63 ff   57
5268 : e0 00 00 00 77 ff c0 00   c3
5270 : 00 00 38 00 80 00 00 00   87
5278 : 31 ff 00 00 00 00 26 00   42
5280 : 00 00 00 00 28 00 00 00   03
5288 : 00 00 50 00 00 00 00 00   9d
5290 : 80 00 00 00 00 00 00 00   11
5298 : 00 00 00 00 00 00 00 00   99
52a0 : 00 00 00 00 00 00 00 00   a1
52a8 : 00 00 00 00 00 00 00 00   a9
52b0 : 00 00 1b 05 00 00 00 00   18
52b8 : 00 00 20 00 00 00 00 00   c1
52c0 : 40 00 01 00 00 00 c0 00   44
52c8 : 01 00 00 00 c0 00 02 00   de
52d0 : 00 00 61 00 06 00 00 00   89
52d8 : 63 ff fc 00 00 00 77 ff   58
52e0 : f8 00 00 00 38 00 10 00   9c
52e8 : 00 00 31 ff c0 00 00 00   41
52f0 : 26 00 00 00 00 00 28 00   b7
52f8 : 00 00 00 00 50 00 00 00   fe
5300 : 00 00 80 00 00 00 00 00   21
5308 : 00 00 00 00 00 00 00 00   09
5310 : 00 00 00 00 00 00 00 00   11
5318 : 00 00 00 00 00 00 00 00   19
5320 : 00 00 00 00 1b 05 20 00   7b
5328 : 10 00 00 00 40 00 30 00   fd
5330 : 00 00 c0 00 70 00 00 00   68
5338 : c7 ff b0 00 00 00 6f ff   e9
5340 : 30 00 00 00 70 00 30 00   38
5348 : 00 00 40 00 30 00 00 00   5c
5350 : ff ff f0 00 00 00 10 40   4c
5358 : 20 00 00 00 20 c0 00 00   81
5360 : 00 00 41 80 00 00 00 00   c1
5368 : e2 80 00 00 00 00 74 c0   de
5370 : 00 00 00 00 38 c0 00 00   fa
5378 : 00 00 10 ff e0 00 00 00   8a
5380 : 00 7f f0 00 00 00 00 00   7c
5388 : 20 00 00 00 00 00 00 00   a9
5390 : 00 00 00 00 00 00 21 05   1f
5398 : 20 00 10 00 00 00 40 00   be
53a0 : 20 00 00 00 c4 00 60 00   8e
53a8 : 00 00 cf ff e0 00 00 00   aa
53b0 : 7f ff 60 00 00 00 60 02   cd
53b8 : 60 00 00 00 60 00 60 00   a0
53c0 : 00 00 7f ff e0 00 00 00   ae
53c8 : 60 00 60 00 00 00 40 00   42
53d0 : 20 00 00 00 80 00 30 00   b9
53d8 : 00 00 00 00 30 00 00 00   dc
53e0 : 00 00 30 00 00 00 00 00   ed
53e8 : 30 00 00 00 00 00 20 00   99
53f0 : 00 00 00 00 40 00 00 00   f5
53f8 : 00 00 00 00 00 00 00 00   f9
5400 : 00 00 00 00 00 00 00 00   01
5408 : 1f 05 08 00 10 00 00 00   ad
5410 : 1f ff f0 00 00 00 3f ff   68
5418 : f0 00 00 00 00 00 20 00   89
5420 : 00 00 7f ff 80 00 00 00   08
5428 : c0 00 10 00 00 00 c0 00   f0
5430 : 30 00 00 00 7f ff f0 00   1c
5438 : 00 00 3f ff f0 00 00 00   17
5440 : 00 00 30 00 00 00 7f ff   4a
5448 : f0 00 00 00 c1 08 20 00   15
5450 : 00 00 c2 10 00 00 00 00   03
5458 : c4 20 10 00 00 00 e8 40   55
5460 : 30 00 00 00 70 80 70 00   5d
5468 : 00 80 3f ff 30 00 00 00   7b
5470 : 0f fc 20 00 00 00 03 f0   f3
5478 : 40 00 25 05 00 00 10 00   e3
5480 : 00 00 08 00 30 00 00 00   86
5488 : 1f ff f0 00 00 00 3f ff   e0
5490 : f0 00 00 00 00 00 30 00   41
5498 : 00 00 7f ff f0 00 00 00   87
54a0 : c1 08 20 00 00 00 c2 10   99
54a8 : 00 00 00 00 c4 20 10 00   36
54b0 : 00 00 e8 40 30 00 00 00   f6
54b8 : 70 80 70 00 00 80 3f ff   85
54c0 : 30 00 00 00 0f fc 20 00   4a
54c8 : 00 00 03 f0 40 00 00 00   ab
54d0 : 00 00 00 00 00 00 00 00   d1
54d8 : 00 00 00 00 00 00 00 00   d9
54e0 : 00 00 00 00 00 00 00 00   e1
54e8 : 00 00 00 00 1b 05 03 fc   c8
54f0 : 00 00 00 00 0f ff 00 00   e1
54f8 : 00 00 10 07 80 00 00 00   e5
5500 : 20 03 c0 00 00 00 40 01   d5
5508 : e0 00 00 00 7f fe e0 00   5c
5510 : 00 00 ff fc f0 00 00 00   bf
5518 : 80 00 70 00 00 00 80 00   b7
5520 : 70 00 00 00 ff ff f0 00   54
5528 : 00 00 e1 10 30 00 00 00   a6
5530 : e1 10 30 00 00 00 f1 10   0d
5538 : 20 00 00 00 71 10 60 00   72
5540 : 00 00 71 10 40 00 00 00   a3
5548 : 39 10 80 00 00 00 1d 11   40
5550 : 00 00 00 00 0f 96 00 00   f6
5558 : 00 00 03 f8 00 00 25 05   d7
5560 : 20 00 20 00 00 00 40 00   8a
5568 : 41 00 00 00 c4 00 c3 00   05
5570 : 00 00 cf ff fb 00 00 00   24
5578 : df ff f3 00 00 00 c0 00   57
5580 : c3 00 00 00 60 00 c3 00   59
5588 : 00 00 7f ff ff 00 00 00   68
5590 : 60 22 c3 00 00 00 c0 44   7e
5598 : 42 00 00 00 c0 88 60 00   ac
55a0 : 00 00 e1 10 20 00 00 00   1d
55a8 : 62 20 20 00 00 00 74 40   75
55b0 : 60 00 00 00 3f ff e0 00   88
55b8 : 00 00 1f ff c0 00 00 00   8c
55c0 : 07 ff 00 00 00 00 00 00   c7
55c8 : 00 00 00 00 00 00 00 00   c9
55d0 : 21 05 03 fc 00 00 00 00   d4
55d8 : 0f ff 00 00 00 00 10 07   35
```

```
55e0 : 80 00 00 00 20 03 c0 00   7e
55e8 : 00 00 40 01 e0 00 00 00   27
55f0 : 7f fe e0 00 00 00 ff fc   20
55f8 : f0 00 00 00 80 00 70 00   b2
5600 : 00 00 80 00 70 00 00 00   28
5608 : ff ff f0 00 00 00 e1 10   eb
5610 : 30 00 00 00 e1 10 70 00   a1
5618 : 00 00 f1 10 60 00 00 00   9d
5620 : 71 10 e0 00 00 00 71 10   b7
5628 : c0 00 00 00 39 11 e0 00   88
5630 : 00 00 1d 11 f0 00 00 00   a9
5638 : 0f 96 e0 00 00 00 03 f8   c8
5640 : 40 00 25 05 20 00 10 00   ad
5648 : 00 00 40 00 30 00 00 00   5c
5650 : c0 00 70 00 00 00 c7 ff   4c
5658 : b0 00 00 00 6f ff 30 00   c0
5660 : 00 00 70 00 30 00 00 00   80
5668 : 40 00 30 00 00 00 ff ff   b4
5670 : f0 00 00 00 10 80 20 00   e6
5678 : 00 00 60 80 00 00 00 00   a1
5680 : c0 80 00 00 00 00 c1 80   89
5688 : 00 00 00 00 c3 c0 00 00   cb
5690 : 00 00 7f 70 00 00 00 00   7e
5698 : 7f 3f c0 00 00 00 3e 3f   5e
56a0 : e0 00 00 00 00 0f f0 00   bd
56a8 : 00 00 00 00 20 00 00 00   ab
56b0 : 00 00 00 00 23 05 00 01   0d
56b8 : c0 00 00 00 00 03 e0 00   14
56c0 : 00 00 00 07 90 00 00 00   aa
56c8 : 03 8f 10 00 00 00 04 cf   47
56d0 : 10 00 00 00 38 4f 10 00   1f
56d8 : 00 00 7c 5f 10 00 00 00   e4
56e0 : 9e 67 90 00 00 00 8f 73   7b
56e8 : e0 00 00 00 cf b1 e0 00   d6
56f0 : 00 00 c7 b0 f0 00 00 00   07
56f8 : c7 98 f0 00 00 00 e7 99   1a
5700 : 70 00 00 00 7f 8e 70 00   9f
5708 : 00 00 77 80 f0 00 00 00   05
5710 : 23 c0 e0 00 00 00 43 c1   5c
5718 : e0 00 00 00 81 e3 c0 00   33
5720 : 00 00 00 7f 00 00 25 05   af
5728 : 21 fc 00 00 00 00 47 ff   e5
5730 : 00 00 00 00 c8 0f 80 00   37
5738 : 00 00 d0 03 c0 00 00 00   d9
5740 : e0 01 e0 00 00 00 ff fe   d7
5748 : e0 00 00 00 7f fc 70 00   ca
5750 : 00 00 60 00 70 00 00 00   70
5758 : 60 00 30 00 00 00 7f ff   c2
5760 : f0 00 00 00 60 00 10 00   97
5768 : 00 00 c0 00 30 00 00 00   9c
5770 : c0 00 20 00 00 00 c0 00   3c
5778 : 60 00 00 00 c0 00 40 00   e6
5780 : 00 00 c0 00 80 00 00 00   b9
5788 : 80 01 00 00 00 00 00 00   89
5790 : 00 00 00 00 00 00 00 00   91
5798 : 21 05 40 7f 00 00 00 00   3c
57a0 : 81 ff c0 00 00 00 87 ff   6f
57a8 : e0 00 00 00 cc 00 70 00   17
57b0 : 00 00 70 00 30 00 00 00   d0
57b8 : 60 00 30 00 00 00 60 00   a6
57c0 : 30 00 00 00 7f ff d0 00   2c
57c8 : 00 00 40 90 10 00 00 00   ec
57d0 : 80 90 20 00 00 00 00 90   c2
57d8 : 40 00 00 00 3f ff c0 00   0f
57e0 : 00 00 7f ff e0 00 00 00   ce
57e8 : 80 90 70 00 00 00 80 00   cf
57f0 : 20 00 00 00 00 00 00 00   11
57f8 : 00 00 00 00 00 00 00 00   f9
5800 : 00 00 00 00 00 00 00 00   01
5808 : 00 00 1d 05 40 10 10 00   b5
5810 : 00 00 80 10 20 00 00 00   35
5818 : ff ff e0 00 00 00 7f ff   4d
5820 : e0 00 00 00 7f ff f0 00   bc
5828 : 00 00 80 90 30 00 00 00   5e
5830 : 00 90 30 00 00 00 00 90   a6
5838 : 10 00 00 00 00 90 10 00   0d
5840 : 00 00 40 90 30 00 00 00   66
5848 : 80 90 f0 00 00 00 7f ff   4a
5850 : e0 00 00 00 1f ff c0 00   25
5858 : 00 00 07 ff 00 00 00 00   1a
5860 : 00 00 00 00 00 00 00 00   61
5868 : 00 00 00 00 00 00 00 00   69
5870 : 00 00 00 00 00 00 00 00   71
5878 : 00 00 00 00 1b 05 80 80   55
5880 : 10 00 00 00 7f ff e0 00   0c
5888 : 00 00 3f ff c0 00 00 00   64
5890 : 3f ff 40 00 00 00 40 90   01
5898 : 60 00 00 00 80 90 30 00   46
58a0 : 00 00 80 90 30 00 00 00   d6
58a8 : ff ff e0 00 00 00 7f ff   dd
58b0 : c0 00 00 00 3f ff 40 00   65
58b8 : 00 00 40 90 60 00 00 00   e1
58c0 : 80 90 30 00 00 00 80 90   b8
58c8 : 30 00 00 00 ff ff e0 00   7c
58d0 : 00 00 7f ff c0 00 00 00   bc
58d8 : 3f ff 00 00 00 00 00 00   17
58e0 : 00 00 00 00 00 00 00 00   e1
58e8 : 00 00 00 00 00 00 1f 05   6f
58f0 : 20 80 04 00 00 08 41 80   98
58f8 : 10 00 00 20 e1 80 40 00   30
5900 : 00 00 7f ff e0 00 00 00   ee
5908 : 3f ff f0 00 00 00 01 80   88
5910 : 20 00 00 00 3f fe 00 00   1c
5918 : 00 00 41 80 00 00 00 00   79
5920 : e1 80 00 00 00 00 71 00   07
5928 : 00 00 00 00 20 00 00 00   2b
5930 : 00 00 00 00 00 00 00 00   31
5938 : 00 00 00 00 00 00 00 00   39
5940 : 00 00 00 00 00 00 00 00   41
5948 : 00 00 00 00 00 00 00 00   49
5950 : 00 00 00 00 00 00 00 00   51
5958 : 00 00 00 00 00 00 00 00   59
5960 : 15 05 20 00 00 00 00 00   00
5968 : 40 00 10 00 00 00 e0 00   30
5970 : 30 00 00 00 7f ff f0 00   5c
5978 : 00 00 3f ff f0 00 00 00   57
5980 : 00 00 30 00 00 00 3f ff   89
5988 : f0 00 00 00 40 00 20 00   fd
5990 : 00 00 c0 00 22 00 00 00   e3
5998 : c0 00 44 00 00 00 c0 00   6d
59a0 : 4c 00 00 00 ff ff f8 00   d0
59a8 : 00 00 7f ff f0 00 00 00   97
59b0 : 40 00 00 00 00 00 00 00   f1
59b8 : 00 00 00 00 00 00 00 00   b9
59c0 : 00 00 00 00 00 00 00 00   c1
59c8 : 00 00 00 00 00 00 00 00   c9
59d0 : 00 00 1b 05 10 00 08 00   59
59d8 : 00 00 20 00 18 00 00 00   62
59e0 : 60 00 30 00 00 00 e0 42   55
59e8 : 70 00 00 00 e0 cc b0 00   90
59f0 : 00 00 60 d9 30 00 00 00   47
59f8 : 60 f2 30 00 00 00 60 e4   29
5a00 : 30 00 00 00 60 c8 30 00   3e
5a08 : 00 00 61 d0 30 00 00 00   7e
5a10 : 63 e0 30 00 00 00 66 c0   0b
5a18 : 30 00 00 00 6c c0 30 00   d6
5a20 : 00 00 79 c0 38 00 00 00   1a
5a28 : 72 c0 38 00 00 00 64 c0   1c
5a30 : 30 00 00 00 c8 80 20 00   72
5a38 : 00 00 80 00 40 00 00 00   5d
5a40 : 00 00 00 00 23 05 00 00   9b
5a48 : 00 00 00 00 00 00 00 00   49
5a50 : 00 00 00 00 00 00 00 00   51
5a58 : 00 00 00 00 00 00 00 00   59
5a60 : 00 00 00 00 00 00 00 00   61
5a68 : 00 00 00 00 00 00 00 00   69
5a70 : 00 00 00 00 00 00 00 00   71
5a78 : 00 00 00 00 00 00 00 00   79
5a80 : 00 00 00 00 00 00 00 00   81
5a88 : 00 00 00 00 00 00 00 00   89
5a90 : 00 00 00 00 00 00 00 00   91
5a98 : 00 00 00 00 00 00 00 00   99
5aa0 : 00 00 00 00 00 00 00 00   a1
5aa8 : 00 00 00 00 00 00 00 00   a9
5ab0 : 00 00 00 00 00 00 00 00   b1
5ab8 : 00 00 00 00 00 00 00 00   b9
5ac0 : 00 00 00 00 00 00 00 00   c1
5ac8 : 00 00 00 00 00 00 00 00   c9
5ad0 : 00 00 00 00 00 00 00 00   d1
5ad8 : 00 00 00 00 00 00 00 00   d9
5ae0 : 00 00 00 00 00 00 00 00   e1
5ae8 : 00 00 00 00 00 00 00 00   e9
5af0 : 00 00 00 00 00 00 00 00   f1
5af8 : 00 00 00 00 00 00 00 00   f9
5b00 : 00 00 00 00 00 00 00 00   01
5b08 : 00 00 00 00 00 00 00 00   09
5b10 : 00 00 00 00 00 00 00 00   11
5b18 : 00 00 00 00 00 00 00 00   19
5b20 : 00 00 00 00 00 00 00 00   21
5b28 : 00 00 00 00 00 00 00 00   29
5b30 : 00 00 00 00 00 00 00 00   31
5b38 : 00 00 00 00 00 00 00 00   39
5b40 : 00 00 00 00 00 00 00 00   41
5b48 : 00 00 00 00 00 00 00 00   49
5b50 : 00 00 00 00 00 00 00 00   51
5b58 : 00 00 00 00 00 00 00 00   59
5b60 : 00 00 00 00 00 00 00 00   61
5b68 : 00 00 00 00 00 00 00 00   69
5b70 : 00 00 00 00 00 00 00 00   71
5b78 : 00 00 00 00 00 00 00 00   79
5b80 : 00 00 00 00 00 00 00 00   81
5b88 : 00 00 00 00 00 00 00 00   89
5b90 : 00 00 00 00 00 00 00 00   91
5b98 : 00 00 00 00 00 00 00 00   99
5ba0 : 00 00 00 00 00 00 00 00   a1
5ba8 : 00 00 00 00 00 00 00 00   a9
5bb0 : 00 00 00 00 00 00 00 00   b1
5bb8 : 00 00 00 00 00 00 00 00   b9
5bc0 : 00 00 00 00 00 00 00 00   c1
5bc8 : 00 00 00 00 00 00 00 00   c9
5bd0 : 00 00 00 00 00 00 00 00   d1
5bd8 : 00 00 00 00 00 00 00 00   d9
5be0 : 00 00 00 00 00 00 00 00   e1
5be8 : 00 00 00 00 00 00 00 00   e9
5bf0 : 00 00 00 00 00 00 00 00   f1
5bf8 : 00 00 00 00 00 00 00 00   f9
5c00 : 00 00 00 00 00 00 00 00   01
5c08 : 00 00 00 00 00 00 00 00   09
5c10 : 00 00 00 00 00 00 00 00   11
5c18 : 00 00 00 00 00 00 00 00   19
5c20 : 00 00 00 00 00 00 00 00   21
5c28 : 00 00 00 00 00 00 00 00   29
5c30 : 00 00 00 00 00 00 00 00   31
5c38 : 00 00 00 00 00 00 00 00   39
5c40 : 00 00 00 00 00 00 00 00   41
5c48 : 00 00 00 00 00 00 00 00   49
5c50 : 00 00 00 00 00 00 00 00   51
5c58 : 00 00 00 00 00 00 00 00   59
5c60 : 00 00 00 00 00 00 00 00   61
5c68 : 00 00 00 00 00 00 00 00   69
5c70 : 00 00 00 00 00 00 00 00   71
5c78 : 00 00 00 00 00 00 00 00   79
5c80 : 00 00 00 00 00 00 00 00   81
5c88 : 00 00 00 00 00 00 00 00   89
5c90 : 00 00 00 00 00 00 00 00   91
5c98 : 00 00 00 00 00 00 00 00   99
5ca0 : 00 00 00 00 00 00 00 00   a1
5ca8 : 00 00 00 00 00 00 00 00   a9
5cb0 : 00 00 00 00 00 00 00 00   b1
5cb8 : 00 00 00 00 00 00 00 00   b9
5cc0 : 00 00 00 00 00 00 00 00   c1
5cc8 : 00 00 00 00 00 00 00 00   c9
```

Listing 3. (Fortsetzung)

```
5cd0 : 00 00 00 00 00 00 00 00  d1
5cd8 : 00 00 00 00 00 00 00 00  d9
5ce0 : 00 00 00 00 00 00 00 00  e1
5ce8 : 00 00 00 00 00 00 00 00  e9
5cf0 : 00 00 00 3f c0 00 00 00  e4
5cf8 : 02 7f e0 00 00 00 04 e0  c4
5d00 : 30 00 00 00 08 c0 10 00  f7
5d08 : 00 00 08 c0 10 00 00 00  24
5d10 : 0c 60 20 00 00 00 06 20  ad
5d18 : 40 00 00 00 03 ff e0 00  0c
5d20 : 00 00 01 ff f0 00 00 00  70
5d28 : 00 00 20 00 00 00 00 00  31
5d30 : 00 00 00 00 00 00 00 00  31
5d38 : 00 00 00 00 00 00 00 00  39
5d40 : 00 00 00 00 00 00 00 00  41
5d48 : 00 00 00 00 00 00 00 00  49
5d50 : 00 00 00 00 00 00 00 00  51
5d58 : 00 00 00 00 00 00 00 00  59
5d60 : 00 00 13 05 20 00 00 00  c8
5d68 : 00 00 40 00 00 00 00 00  79
5d70 : e0 00 40 00 00 00 7f ff  5e
5d78 : e0 00 00 00 3f ff e0 00  d0
5d80 : 00 00 02 00 30 00 00 00  04
5d88 : 04 00 30 00 00 00 0c 00  c9
5d90 : 10 00 00 00 0e 00 20 00  02
5d98 : 00 00 07 ff c0 00 00 00  66
5da0 : 03 ff 80 00 00 00 00 00  c3
5da8 : 00 00 00 00 00 00 00 00  a9
5db0 : 00 00 00 00 00 00 00 00  b1
5db8 : 00 00 00 00 00 00 00 00  b9
5dc0 : 00 00 00 00 00 00 00 00  c1
5dc8 : 00 00 00 00 00 00 00 00  c9
5dd0 : 00 00 00 00 15 05 01 00  4e
5dd8 : 40 00 00 00 03 ff c0 00  4c
5de0 : 00 00 07 ff e0 00 00 00  b0
5de8 : 0e 00 30 00 00 00 0c 00  33
5df0 : 30 00 00 00 06 00 30 00  42
5df8 : 00 00 03 00 20 00 00 00  bb
5e00 : 06 00 40 00 00 00 00 00  17
5e08 : 80 00 00 00 00 00 00 00  89
5e10 : 00 00 00 00 00 00 00 00  11
5e18 : 00 00 00 00 00 00 00 00  19
5e20 : 00 00 00 00 00 00 00 00  21
5e28 : 00 00 00 00 00 00 00 00  29
5e30 : 00 00 00 00 00 00 00 00  31
5e38 : 00 00 00 00 00 00 00 00  39
5e40 : 00 00 00 00 00 00 11 00  85
5e48 : 00 80 40 00 00 00 01 ff  9d
5e50 : c0 00 00 00 63 ff e0 00  ca
5e58 : 00 00 b0 00 60 00 00 00  8b
5e60 : 18 00 30 00 00 00 18 00  e5
5e68 : 30 00 00 00 0c 00 30 00  1a
5e70 : 00 00 06 00 20 00 00 00  f4
5e78 : 07 ff c0 00 00 00 03 ff  bb
5e80 : 80 00 00 00 01 00 00 00  11
5e88 : 00 00 00 00 00 00 00 00  89
5e90 : 00 00 00 00 00 00 00 00  91
5e98 : 00 00 00 00 00 00 00 00  99
5ea0 : 00 00 00 00 00 00 00 00  a1
5ea8 : 00 00 00 00 00 00 00 00  a9
5eb0 : 00 00 00 00 00 00 00 00  b1
5eb8 : 15 05 02 00 40 00 00 00  d5
5ec0 : 03 ff e0 00 00 00 07 ff  17
5ec8 : e0 00 00 00 0c 04 30 00  4a
5ed0 : 00 00 0c 08 30 00 00 00  d8
5ed8 : 06 10 30 00 00 00 03 20  3f
5ee0 : 20 00 00 00 01 c0 40 00  18
5ee8 : 00 00 00 80 00 00 00 00  f9
5ef0 : 00 00 00 00 00 00 00 00  f1
5ef8 : 00 00 00 00 00 00 00 00  f9
5f00 : 00 00 00 00 00 00 00 00  01
5f08 : 00 00 00 00 00 00 00 00  09
5f10 : 00 00 00 00 00 00 00 00  11
5f18 : 00 00 00 00 00 00 00 00  19
5f20 : 00 00 00 00 00 00 00 00  21
5f28 : 00 00 11 05 02 00 00 00  2e
5f30 : 00 00 06 00 40 00 00 00  b6
5f38 : 3f ff e0 00 00 00 7f ff  ad
5f40 : f0 00 00 00 c6 00 20 00  1d
5f48 : 00 00 e4 00 00 00 00 00  82
5f50 : 70 00 00 00 00 00 38 00  a1
5f58 : 00 00 00 00 10 00 00 00  5a
5f60 : 00 00 20 00 00 00 00 00  69
5f68 : 00 00 00 00 00 00 00 00  69
5f70 : 00 00 00 00 00 00 00 00  71
5f78 : 00 00 00 00 00 00 00 00  79
5f80 : 00 00 00 00 00 00 00 00  81
5f88 : 00 00 00 00 00 00 00 00  89
5f90 : 00 00 00 00 00 00 00 00  91
5f98 : 00 00 00 00 13 05 02 01  fc
5fa0 : 01 00 00 00 03 ff 82 00  dc
5fa8 : 00 00 07 ff c6 00 00 00  d7
5fb0 : 04 00 6b 00 00 00 0c 00  bf
5fb8 : 73 00 00 00 0c 00 c3 00  fb
5fc0 : 00 00 07 ff c3 00 00 00  be
5fc8 : 07 ff e2 00 00 00 02 00  90
5fd0 : 1c 00 00 00 00 00 00 00  ed
5fd8 : 00 00 00 00 00 00 00 00  d9
5fe0 : 00 00 00 00 00 00 00 00  e1
5fe8 : 00 00 00 00 00 00 00 00  e9
5ff0 : 00 00 00 00 00 00 00 00  f1
5ff8 : 00 00 00 00 00 00 00 00  f9
6000 : 00 00 00 00 00 00 00 00  01
6008 : 00 00 00 00 00 00 11 05  57
6010 : 40 00 40 00 00 00 ff ff  60
6018 : e0 00 00 00 7f ff f1 00  b8
6020 : 00 00 22 00 21 00 00 00  bb
6028 : 04 00 02 00 00 00 0c 00  dd
6030 : 06 00 00 00 0e 00 0c 00  48
6038 : 00 00 07 ff f8 00 00 00  8a
6040 : 03 ff f0 00 00 00 00 00  7f
6048 : 00 00 00 00 00 00 00 00  49
6050 : 00 00 00 00 00 00 00 00  51
6058 : 00 00 00 00 00 00 00 00  59
6060 : 00 00 00 00 00 00 00 00  61
6068 : 00 00 00 00 00 00 00 00  69
6070 : 00 00 00 00 00 00 00 00  71
6078 : 00 00 00 00 00 00 00 00  79
6080 : 11 05 02 00 40 00 00 00  99
6088 : 67 ff e0 00 00 00 cf ff  66
6090 : f0 00 00 00 c4 00 20 00  4d
6098 : 00 00 00 00 00 00 00 00  99
60a0 : 00 00 00 00 00 00 00 00  a1
60a8 : 00 00 00 00 00 00 00 00  a9
60b0 : 00 00 00 00 00 00 00 00  b1
60b8 : 00 00 00 00 00 00 00 00  b9
60c0 : 00 00 00 00 00 00 00 00  c1
60c8 : 00 00 00 00 00 00 00 00  c9
60d0 : 00 00 00 00 00 00 00 00  d1
60d8 : 00 00 00 00 00 00 00 00  d9
60e0 : 00 00 00 00 00 00 00 00  e1
60e8 : 00 00 00 00 00 00 00 00  e9
60f0 : 00 00 07 05 00 00 02 00  5b
60f8 : 00 00 00 00 04 00 00 00  39
6100 : 02 00 0c 00 00 00 67 ff  a3
6108 : f8 00 00 00 cf ff f8 00  e1
6110 : 00 00 c4 00 10 00 00 00  43
6118 : 00 00 00 00 00 00 00 00  19
6120 : 00 00 00 00 00 00 00 00  21
6128 : 00 00 00 00 00 00 00 00  29
6130 : 00 00 00 00 00 00 00 00  31
6138 : 00 00 00 00 00 00 00 00  39
6140 : 00 00 00 00 00 00 00 00  41
6148 : 00 00 00 00 00 00 00 00  49
6150 : 00 00 00 00 00 00 00 00  51
6158 : 00 00 00 00 00 00 00 00  59
6160 : 00 00 00 00 0b 05 20 00  ba
6168 : 40 00 00 00 7f ff e0 00  24
6170 : 00 00 ff ff f0 00 00 00  7f
6178 : 42 10 20 00 00 00 04 20  1b
6180 : 00 00 00 00 0c 40 00 00  43
6188 : 00 00 0e c0 40 00 00 00  28
6190 : 07 3f e0 00 00 00 02 1f  b5
6198 : f0 00 00 00 00 00 20 00  09
61a0 : 00 00 00 00 00 00 00 00  a1
61a8 : 00 00 00 00 00 00 00 00  a9
61b0 : 00 00 00 00 00 00 00 00  b1
61b8 : 00 00 00 00 00 00 00 00  b9
61c0 : 00 00 00 00 00 00 00 00  c1
61c8 : 00 00 00 00 00 00 00 00  c9
61d0 : 00 00 00 00 00 00 13 05  27
61d8 : 40 00 80 00 00 00 ff ff  38
61e0 : c0 00 00 00 7f ff e0 00  1c
61e8 : 00 00 20 00 70 00 00 00  f8
61f0 : 00 00 20 00 00 00 00 00  f9
61f8 : 40 00 00 00 00 00 00 00  39
6200 : 00 00 00 00 00 00 00 00  01
6208 : 00 00 00 00 00 00 00 00  09
6210 : 00 00 00 00 00 00 00 00  11
6218 : 00 00 00 00 00 00 00 00  19
6220 : 00 00 00 00 00 00 00 00  21
6228 : 00 00 00 00 00 00 00 00  29
6230 : 00 00 00 00 00 00 00 00  31
6238 : 00 00 00 00 00 00 00 00  39
6240 : 00 00 00 00 00 00 00 00  41
6248 : 0b 05 04 00 40 00 00 00  db
6250 : 0f ff e0 00 00 00 07 ff  b3
6258 : f0 00 00 00 02 00 20 00  e9
6260 : 00 00 06 00 00 00 00 00  e2
6268 : 04 00 00 00 00 00 0c 00  9d
6270 : 40 00 00 00 0f ff e0 00  25
6278 : 00 00 07 ff f0 00 00 00  49
6280 : 02 00 20 00 00 00 06 00  a3
6288 : 00 00 00 00 04 00 00 00  c9
6290 : 00 00 0c 00 40 00 00 00  98
6298 : 0f ff e0 00 00 00 07 ff  fb
62a0 : f0 00 00 00 02 00 20 00  31
62a8 : 00 00 00 00 00 00 00 00  a9
62b0 : 00 00 00 00 00 00 00 00  b1
62b8 : 00 00 1f 05 04 00 40 00  62
62c0 : 00 00 0f ff e0 00 00 00  92
62c8 : 07 ff f0 00 00 00 02 00  13
62d0 : 20 00 00 00 06 00 00 00  51
62d8 : 00 00 04 00 00 00 00 00  da
62e0 : 0c 00 40 00 00 00 0f ff  39
62e8 : e0 00 00 00 07 ff f0 00  fd
62f0 : 00 00 00 00 20 00 00 00  f3
62f8 : 00 00 00 00 00 00 00 00  f9
6300 : 00 00 00 00 00 00 00 00  01
6308 : 00 00 00 00 00 00 00 00  09
6310 : 00 00 00 00 00 00 00 00  11
6318 : 00 00 00 00 00 00 00 00  19
6320 : 00 00 00 00 00 00 00 00  21
6328 : 00 00 00 00 13 05 02 00  8a
6330 : 40 00 00 00 03 ff c0 00  a4
6338 : 00 00 07 ff e0 00 00 00  08
6340 : 04 00 60 00 00 00 0c 00  8d
6348 : 30 00 00 00 0c 00 30 00  fa
6350 : 00 00 06 00 30 00 00 00  d5
6358 : 07 ff e0 00 00 00 03 ff  a3
6360 : c0 00 00 00 00 00 00 00  21
6368 : 00 00 00 00 00 00 00 00  69
6370 : 00 00 00 00 00 00 00 00  71
6378 : 00 00 00 00 00 00 00 00  79
6380 : 00 00 00 00 00 00 00 00  81
6388 : 00 00 00 00 00 00 00 00  89
6390 : 00 00 00 00 00 00 00 00  91
6398 : 00 00 00 00 00 00 11 05  e7
63a0 : 02 00 00 00 00 00 04 00  b3
63a8 : 01 00 00 00 0e 00 43 00  97
63b0 : 00 00 07 ff fe 00 00 00  62
63b8 : 03 ff fc 00 00 00 02 00  02
63c0 : 20 00 00 00 04 00 10 00  61
63c8 : 00 00 0c 00 30 00 00 00  cf
63d0 : 0e 00 60 00 00 00 07 ff  13
```

```
63d8 : c0 00 00 00 03 ff 80 00  cb
63e0 : 00 00 00 00 00 00 00 00  e1
63e8 : 00 00 00 00 00 00 00 00  e9
63f0 : 00 00 00 00 00 00 00 00  f1
63f8 : 00 00 00 00 00 00 00 00  f9
6400 : 00 00 00 00 00 00 00 00  01
6408 : 00 00 00 00 00 00 00 00  09
6410 : 15 05 03 ff c0 00 00 00  75
6418 : 07 ff e0 00 00 00 0c 00  87
6420 : 20 00 00 00 0c 00 11 00  46
6428 : 00 00 06 00 11 00 00 00  bb
6430 : 02 00 32 00 00 00 03 ff  cb
6438 : fe 00 00 00 07 ff fc 00  9b
6440 : 00 00 0e 00 20 00 00 00  c6
6448 : 04 00 00 00 00 00 02 00  55
6450 : 00 00 00 00 00 00 00 00  51
6458 : 00 00 00 00 00 00 00 00  59
6460 : 00 00 00 00 00 00 00 00  61
6468 : 00 00 00 00 00 00 00 00  69
6470 : 00 00 00 00 00 00 00 00  71
6478 : 00 00 00 00 00 00 00 00  79
6480 : 00 00 15 05 04 00 40 00  a8
6488 : 00 00 0f ff e0 00 00 00  5a
6490 : 07 ff f0 00 00 00 02 00  db
6498 : 20 00 00 00 04 00 00 00  f9
64a0 : 00 00 0e 00 00 00 00 00  24
64a8 : 07 00 00 00 00 00 02 00  b8
64b0 : 00 00 00 00 00 00 00 00  b1
64b8 : 00 00 00 00 00 00 00 00  b9
64c0 : 00 00 00 00 00 00 00 00  c1
64c8 : 00 00 00 00 00 00 00 00  c9
64d0 : 00 00 00 00 00 00 00 00  d1
64d8 : 00 00 00 00 00 00 00 00  d9
64e0 : 00 00 00 00 00 00 00 00  e1
64e8 : 00 00 00 00 00 00 00 00  e9
64f0 : 00 00 00 00 0f 05 00 00  0a
64f8 : 10 00 00 00 03 e0 20 00  c0
6500 : 00 40 07 f0 60 00 00 80  07
6508 : 0e 19 60 00 00 00 0c 0a  ff
6510 : 60 00 04 00 0c 08 30 00  33
6518 : 10 00 0c 20 30 00 50 00  74
6520 : 06 00 30 00 90 00 07 08  68
6528 : 70 00 00 00 06 0f e0 04  fd
6530 : 00 00 00 07 c0 08 00 00  5e
6538 : 00 00 00 00 00 00 00 00  39
6540 : 00 00 00 00 00 00 00 00  41
6548 : 00 00 00 00 00 00 00 00  49
6550 : 00 00 00 00 00 00 00 00  51
6558 : 00 00 00 00 00 00 00 00  59
6560 : 00 00 00 00 00 00 16 05  c3
6568 : 08 00 00 00 00 00 18 00  d1
6570 : 80 00 00 00 3f ff c0 00  e7
6578 : 00 00 ff ff e0 00 00 00  86
6580 : 08 00 70 00 00 00 08 00  c5
6588 : 20 00 00 00 00 00 40 00  aa
6590 : 00 00 00 00 00 00 00 00  91
6598 : 00 00 00 00 00 00 00 00  99
65a0 : 00 00 00 00 00 00 00 00  a1
65a8 : 00 00 00 00 00 00 00 00  a9
65b0 : 00 00 00 00 00 00 00 00  b1
65b8 : 00 00 00 00 00 00 00 00  b9
65c0 : 00 00 00 00 00 00 00 00  c1
65c8 : 00 00 00 00 00 00 00 00  c9
65d0 : 00 00 00 00 00 00 00 00  d1
65d8 : 0d 05 02 00 40 00 00 00  ed
65e0 : 07 ff e0 00 00 00 0f ff  5b
65e8 : e0 00 00 00 04 00 70 00  cb
65f0 : 00 00 00 00 20 00 00 00  f3
65f8 : 00 00 40 00 00 00 02 00  11
6600 : 80 00 00 00 07 ff c0 00  f4
6608 : 00 00 0f ff e0 00 00 00  da
6610 : 04 00 70 00 00 00 00 00  31
6618 : 20 00 00 00 00 00 00 00  39
6620 : 00 00 00 00 00 00 00 00  21
6628 : 00 00 00 00 00 00 00 00  29
6630 : 00 00 00 00 00 00 00 00  31
6638 : 00 00 00 00 00 00 00 00  39
6640 : 00 00 00 00 00 00 00 00  41
6648 : 00 00 15 05 18 00 40 00  b1
6650 : 00 00 2f ff e0 00 00 00  2a
6658 : 07 ff e0 00 00 00 03 00  a3
6660 : 70 00 00 00 00 00 30 00  91
6668 : 00 00 04 00 20 00 00 00  6c
6670 : 0e 00 60 00 00 00 07 ff  b3
6678 : c0 00 00 00 03 ff 00 00  69
6680 : 00 00 01 00 00 00 00 00  c1
6688 : 00 00 00 00 00 00 00 00  89
6690 : 00 00 00 00 00 00 00 00  91
6698 : 00 00 00 00 00 00 00 00  99
66a0 : 00 00 00 00 00 00 00 00  a1
66a8 : 00 00 00 00 00 00 00 00  a9
66b0 : 00 00 00 00 00 00 00 00  b1
66b8 : 00 00 00 00 13 05 18 00  72
66c0 : 40 00 00 00 2f ff e0 00  77
66c8 : 00 00 07 ff e0 00 00 00  98
66d0 : 03 00 70 00 00 00 00 00  f0
66d8 : 30 00 00 00 04 00 20 00  c9
66e0 : 00 00 0e 00 20 00 00 00  66
66e8 : 07 ff c0 00 00 00 03 ff  2b
66f0 : e0 00 00 00 00 00 70 00  92
66f8 : 00 00 04 00 30 00 00 00  fd
6700 : 0e 00 20 00 00 00 07 ff  33
6708 : e0 00 00 00 03 ff c0 00  1c
6710 : 00 00 01 00 40 00 00 00  55
6718 : 00 00 00 00 00 00 00 00  19
6720 : 00 00 00 00 00 00 00 00  21
6728 : 00 00 00 00 00 00 1d 05  a7
6730 : 00 20 04 00 00 08 04 60  53
6738 : 90 00 00 20 0f ff c0 00  c0
6740 : 00 00 07 ff e0 00 00 00  10
6748 : 02 60 70 00 00 00 04 60  67
6750 : 20 00 00 00 0e 40 00 00  53
6758 : 00 00 07 00 00 00 00 00  1a
6760 : 02 00 00 00 00 00 00 00  63
6768 : 00 00 00 00 00 00 00 00  69
6770 : 00 00 00 00 00 00 00 00  71
6778 : 00 00 00 00 00 00 00 00  79
6780 : 00 00 00 00 00 00 00 00  81
6788 : 00 00 00 00 00 00 00 00  89
6790 : 00 00 00 00 00 00 00 00  91
6798 : 00 00 00 00 00 00 00 00  99
67a0 : 11 05 04 00 80 00 00 00  3d
67a8 : 0f ff c0 00 00 00 07 ff  03
67b0 : e0 00 00 00 02 00 72 00  7a
67b8 : 00 00 06 00 22 00 00 00  5c
67c0 : 04 00 04 00 00 00 0c 00  f6
67c8 : 0c 00 00 00 0f ff f8 00  a9
67d0 : 00 00 07 ff f0 00 00 00  a1
67d8 : 00 00 00 00 00 00 00 00  d9
67e0 : 00 00 00 00 00 00 00 00  e1
67e8 : 00 00 00 00 00 00 00 00  e9
67f0 : 00 00 00 00 00 00 00 00  f1
67f8 : 00 00 00 00 00 00 00 00  f9
6800 : 00 00 00 00 00 00 00 00  01
6808 : 00 00 00 00 00 00 00 00  09
6810 : 00 00 11 05 01 00 1c 00  76
6818 : 00 00 02 04 62 00 00 00  40
6820 : 04 0c 84 00 00 00 0c 1e  b8
6828 : 0c 00 00 00 07 e7 f8 00  c8
6830 : 00 00 03 c3 f0 00 00 00  79
6838 : 00 00 00 00 00 00 00 00  39
6840 : 00 00 00 00 00 00 00 00  41
6848 : 00 00 00 00 00 00 00 00  49
6850 : 00 00 00 00 00 00 00 00  51
6858 : 00 00 00 00 00 00 00 00  59
6860 : 00 00 00 00 00 00 00 00  61
6868 : 00 00 00 00 00 00 00 00  69
6870 : 00 00 00 00 00 00 00 00  71
6878 : 00 00 00 00 00 00 00 00  79
6880 : 00 00 00 00 0b 05 00 3f  d8
6888 : c0 00 00 00 02 7f e0 c0  6a
6890 : 00 00 04 e0 30 60 00 00  b4
6898 : 08 c0 10 00 00 00 08 c0  a6
68a0 : 10 00 00 00 0c 60 20 c0  76
68a8 : 00 00 06 20 40 60 00 00  35
68b0 : 03 ff e0 00 00 00 01 ff  ef
68b8 : f0 00 00 00 00 00 20 00  29
68c0 : 00 00 00 00 00 00 00 00  c1
68c8 : 00 00 00 00 00 00 00 00  c9
68d0 : 00 00 00 00 00 00 00 00  d1
68d8 : 00 00 00 00 00 00 00 00  d9
68e0 : 00 00 00 00 00 00 00 00  e1
68e8 : 00 00 00 00 00 00 00 00  e9
68f0 : 00 00 00 00 00 00 13 05  47
68f8 : 02 00 40 00 00 00 03 ff  17
6900 : c0 c0 00 00 07 ff e0 60  d5
6908 : 00 00 04 00 60 00 00 00  10
6910 : 0c 00 30 00 00 00 0c 00  59
6918 : 30 c0 00 00 06 00 30 60  8a
6920 : 00 00 07 ff e0 00 00 00  f0
6928 : 03 ff c0 00 00 00 00 00  5b
6930 : 00 00 00 00 00 00 00 00  31
6938 : 00 00 00 00 00 00 00 00  39
6940 : 00 00 00 00 00 00 00 00  41
6948 : 00 00 00 00 00 00 00 00  49
6950 : 00 00 00 00 00 00 00 00  51
6958 : 00 00 00 00 00 00 00 00  59
6960 : 00 00 00 00 00 00 00 00  61
6968 : 11 05 02 00 40 00 00 00  81
6970 : 07 ff e0 c0 00 00 0f ff  03
6978 : e0 60 00 00 04 00 70 00  8b
6980 : 00 00 00 00 20 00 00 00  83
6988 : 00 00 40 00 00 00 02 00  a1
6990 : 80 00 00 00 c7 ff c0 00  90
6998 : 00 00 6f ff e0 00 00 00  82
69a0 : 04 00 70 00 00 00 00 00  c1
69a8 : 20 00 00 00 00 00 00 00  c9
69b0 : 00 00 00 00 00 00 00 00  b1
69b8 : 00 00 00 00 00 00 00 00  b9
69c0 : 00 00 00 00 00 00 00 00  c1
69c8 : 00 00 00 00 00 00 00 00  c9
69d0 : 00 00 00 00 00 00 00 00  d1
69d8 : 00 00 15 05 00 00 00 00  be
69e0 : 00 00 00 00 00 00 00 00  e1
69e8 : 00 00 00 00 00 00 00 00  e9
69f0 : 00 00 00 00 00 00 00 00  f1
69f8 : 00 00 00 00 00 00 00 00  f9
6a00 : 00 00 00 00 00 00 00 00  01
6a08 : 00 00 00 00 00 00 00 00  09
6a10 : 00 00 00 00 00 00 00 00  11
6a18 : 00 00 00 00 00 00 00 00  19
6a20 : 00 00 00 00 00 00 00 00  21
6a28 : 00 00 00 00 00 00 00 00  29
6a30 : 00 00 00 00 00 00 00 00  31
6a38 : 00 00 00 00 00 00 00 00  39
6a40 : 00 00 00 00 00 00 00 00  41
6a48 : 00 00 00 00 00 00 00 00  49
6a50 : 00 00 00 00 00 00 00 00  51
6a58 : 00 00 00 00 00 00 00 00  59
6a60 : 00 00 00 00 00 00 00 00  61
6a68 : 00 00 00 00 00 00 00 00  69
6a70 : 00 00 00 00 00 00 00 00  71
6a78 : 00 00 00 00 00 00 00 00  79
6a80 : 00 00 00 00 00 00 00 00  81
6a88 : 00 00 00 00 00 00 00 00  89
6a90 : 00 00 00 00 00 00 00 00  91
6a98 : 00 00 00 00 00 00 00 00  99
6aa0 : 00 00 00 00 00 00 00 00  a1
6aa8 : 00 00 00 00 00 00 00 00  a9
6ab0 : 00 00 00 00 00 00 00 00  b1
6ab8 : 00 00 00 00 00 00 00 00  b9
6ac0 : 20 00 00 00 00 00 00 00  e1
```

Listing 3. (Schluß)

Tausende von Lesern programmieren ihren C 128 und stoßen – manchmal durch Zufall, oft aber auch durch systematisches Probieren – auf Lösungen, die auch Sie interessieren könnten. 49 solcher Tips und Tricks haben wir für Sie gesammelt.

Das C 128

Einige Monate sind verstrichen, seitdem wir Ihnen im Sonderheft 29 die Tip- und Trick-Kiste öffneten. Neue und auch verbesserte Lösungen vielleicht auch Ihrer Probleme haben sich inzwischen bei uns gestapelt. Die besten davon finden Sie nun hier in sieben Rubriken zusammengestellt.

Programmiertips nennt sich die erste davon, die die Tips 1 bis 6 umfaßt. Die INPUT-Mängel des Basic 7.0 finden sich auch noch in einer anderen Rubrik gelöst wieder, hier wird Ihnen ein kurzer und einfacher Weg gezeigt, einen Teil dieser Schönheitsfehler zu vermeiden. Wie man die Tasten des C 128 in seinen Programmen effektiv nutzen kann, wie man auch lange Directories mit stets sichtbarer Kopfzeile auf den Bildschirm bringen kann und anderes mehr finden Sie in diesem ersten Abschnitt gesammelt.

PEEK, POKE und SYS heißt die zweite Rubrik. Zwar sind die C 128-Benutzer der Sucht des POKEns noch nicht so sehr verfallen wie die C 64-Freaks, dennoch: In den Tips 7 bis 9 haben wir für Sie einige dieser Ungeheuer abgedruckt, die das Nackenhaar jedes konservativen Informatikers sträuben.

Größere und kleinere Befehlserweiterungen bilden die dritte Rubrik (Tips 10 bis 15). Angefangen von einer recht umfangreichen Ergänzung des Basic 7.0 um 39 Befehle durch das TNT-Basic bis zur neuen Input-Routine, die keinen Wunsch mehr offenläßt, finden Sie hier Verbesserungen des C 128-Basic.

Der nächste Abschnitt hat mit Grafik im weitesten Sinn zu tun. Die Tips 16 bis 26 umfassen sowohl eine Super-Hardcopy-Routine, als auch verschiedene neue Möglichkeiten des 80-Zeichen-Bildschirms. Außerdem erfahren Sie noch von einer Fußangel, die der CHAR-Befehl dem Basic-Programmierer bereithält.

Fünf Beiträge haben wir unter der Überschrift Floppy-Tips zusammengefaßt (Tips 27 bis 31). Natürlich haben sie alle mit der 1571 zu tun.

Der C 64-Modus des C 128 ist das Thema der Tips 32 bis 38. Wie kann man per Programm unterscheiden, ob ein C 64 oder ein C 128 benutzt wird? Wie kann man den besseren Editor des C 128 bei der Eingabe von C 64-Programmen nutzen? Diese und andere Fragen beantwortet Ihnen dieser Abschnitt.

Zu guter Letzt finden Sie noch eine Rubrik namens Sammelsurium, in der die Tips 39 bis 49 untergebracht sind, die wir nicht so recht in die anderen Abschnitte einsortieren konnten.

Kurzübersicht

So finden Sie leichter Ihre Tips & Tricks

Bevor Sie jetzt ans Ausprobieren gehen, noch eine Bemerkung: Wie Sie sicherlich aus den verschiedenen Ausgaben des 64'er-Magazins und aus den Sonderheften wissen, sind die diversen Baureihen des C 128 wahre Wunder an Kompatibilität. Einige Tips beziehen sich speziell auf die Unterschiede, die den C 128 im wurffesten Blechgehäuse auszeichnen, zum Teil laufen die damit verbundenen Programme nicht auf den C 128-Plastik-Verwandten. Im allgemeinen ist das dann aus der Erklärung zum Programm zu erkennen. Lediglich bei VDC-Window (Listings 22 und 23) ist das nicht erwähnt worden. Um Ihnen das Auffinden der einzelnen Tips zu erleichtern, haben wir in nebenstehender Tabelle, die einzelnen Beiträge nach Themen und Seiten geordnet ist.

(Heimo Ponnath/ef)

1. Quicksort

Listing 1 enthält die Basic 7.0-Version des schnellen Quicksort-Algorithmus. Die Bedienung ist denkbar einfach: Der Unterroutine, die mit GOSUB aufgerufen wird, sind nur wenige Parameter zu übergeben. Die Variable O enthält die Anzahl der zu sortierenden Daten, und das String-Feld A$(...) enthält die zu sortierenden Daten. Weiter müssen im Hauptprogramm zwei numerische Felder LG und RG mit mindestens soviel Elementen dimensioniert werden, wie es Daten gibt, die zu sortieren sind. In Listing 1 wird die Bedienung in den Zeilen 1 bis 8 demonstriert.

(K. H. Guckler/ef)

```
1 DIM A$(10): REM *** ZU SORTIERENDES ARRAY   <QJG>
2 DIM LG(10),RG(10): REM *** LINKE & RECHTE
  GRENZE                                      <5VD>
3 O=10: REM *** MENGE DER FELDER              <SNJ>
4 FOR I=1 TO 10: INPUT A$(I): NEXT : REM ***
  EINGABE 10 FELDER                           <4F3>
5 :                                           <A18>
6 GOSUB 40000: REM *** SORTIEREN              <OMD>
7 :                                           <AHA>
8 FOR I=1 TO 10: PRINT A$(I): NEXT : REM ***
  AUSGABE SORTIERT                            <BB7>
9 :                                           <914>
10 END                                        <BTE>
40000 S=1: LG(1)=1: RG(1)=O: DO : GOSUB 4000
      1: LOOP UNTIL S=0: RETURN               <FDJ>
```

Listing 1. »Quick-Sort C-128« sortiert Stringfelder schnell

-Schatzkästchen

```
40001 L=LG(S): R=RG(S): S=S-1: DO : GOSUB 40
      002: LOOP UNTIL L>=R: RETURN              <FNH>
40002 I=L: J=R: M=(L+R)/2: X$=A$(M): DO : GO
      SUB 40004: LOOP UNTIL I>J: IF I<R THEN
       S=S+1: LG(S)=I: RG(S)=R                  <CFI>
40003 R=J: RETURN                               <EI3>
40004 DO UNTIL A$(I)>=X$: I=I+1: LOOP : DO U
      NTIL A$(J)<=X$: J=J-1: LOOP               <SQT>
40005 IF I<=J THEN W$=A$(I): A$(I)=A$(J): A$
      (J)=W$: I=I+1: J=J-1                      <4P4>
40006 RETURN                                    <2Q7>
```

Listing 1. (Schluß)

2. Directory mit stehender Kopfzeile

Der Window-Befehl des C 128 bietet eine Möglichkeit, die man bei professionellen Programmen angewendet sieht, die aber im Basic 7.0 nicht direkt vorgesehen ist: Beim Anzeigen des Directory ist es oft wünschenswert, daß die erste angezeigte Zeile (Diskname und ID) nicht weggescrollt wird. Mit einem kleinen Trick ist das auch in Basic 7.0 einfach realisierbar: Direkt nach der Definition des Fensters (Window x1, y1, x2, y2, 1), in dem das Directory ausgegeben werden soll, gibt man folgendes ein: POKE 229, PEEK(229)+1 (bzw. in Assembler »INC $E5«). Dadurch wird der obere Rand des Fensters um eine Zeile nach unten verschoben, ohne dabei die Cursorposition zu verändern. Nun kann die Directory-Routine angesprungen werden ('DIRECTORY', bzw. »JSR $A07E«). Die erste Zeile, die nun ausgegeben wird (Diskname etc.), liegt gerade außerhalb des Fensters, wird also nicht mitgescrollt. Auch hierzu nun ein Programmbeispiel:

```
10 REM ** DIRECTORY MIT KOPFZEILE **
20 WINDOW 25,5,70,20,1        :REM WINDOW SETZEN
30 POKE 229,PEEK(229)+1       :REM OBERSTE ZEILE
                               SCHUETZEN
40 DIRECTORY                  :REM DIR AUSGEBEN
```

(H. Stöcklein/ef)

3. Unterscheidung aller Tasten des C 128

In vielen professionellen Programmen sieht man immer wieder, daß Tastenkombinationen abgefragt werden, die mit dem »GET«-Befehl (bzw. mit JSR GETIN) nicht eindeutig abfragbar sind. Dazu zählt die Funktion »CTRL-ENTER« unter CP/M 3.0, die ja dort bekanntlich einen Reset bewirkt.

So etwas ist aber auch in Basic 7.0 und 8502-Assembler ganz einfach möglich.

Dazu verwendet man am besten die Speicherstellen 211 und 212 der Zeropage. Beide Zellen findet man (in vereinfachter Ausführung) auch beim C 64, dort liegen sie allerdings bei 653 und 203. Doch wie ist eine Tastaturabfrage mittels dieser Speicherstellen beim C 128 nun zu bewerkstelligen?

Dazu muß man wissen, daß das Betriebssystem des C 128 die Tastatur (von RESTORE einmal abgesehen) in zwei Sorten von Tasten unterteilt:

A) SHIFT, CBM, CTRL, ALT und ASCII-DIN lassen sich in der Zelle 211 abfragen, und zwar wie folgt: Jeder dieser Tasten ist ein Bit in der erwähnten Speicherstelle zugeordnet, das gesetzt wird, wenn man die Taste betätigt.

211 ($D3):	Bit	4	3	2	1	0
	Wert	16	8	4	2	1
	Taste	ASCII	ALT	CTRL	CBM	SHIFT

Drückt man mehrere dieser Tasten gleichzeitig, so werden auch die betreffenden Bits gleichzeitig gesetzt, man muß also die Wertigkeiten addieren. Ein Beispiel: Drückt man die Tastenkombination <SHIFT CTRL ALT>, so findet man in 211 den Wert 1+4+8 = 13.

B) Alle anderen Tasten werden in der Speicherstelle 212 festgehalten. Um den Wert zu erfahren, den eine gedrückte Taste hier bedeutet, kann man eine Tabelle des C 128-Bedienungshandbuchs verwenden, die im Anhang J (»Definition von Tastaturbelegungen«), abgedruckt ist. Dort sind alle Tasten der Gruppe B in einer Reihenfolge aufgeführt, die weder dem ASCII-Code noch sonst einer Norm entspricht (in dieser Reihenfolge sind die entsprechenden ASCII-Werte in den Tastaturtabellen des Computers abgelegt). Hier sind die abgesetzten Tasten des Zehnerblocks und des Cursorblock eindeutig zu unterscheiden von den »normalen« Tasten der Tastatur! Wichtig ist, daß die Nummer, die bei jeder Taste in der Spalte »Platz« ganz links steht, um eins höher ist als der entsprechende Wert dieser Taste in der Speicherstelle 212. Es gilt also zur Berechnung:

PEEK (212) = Platznummer — 1

Mit Hilfe dieser Speicherstellen lassen sich nun auch exotische Tastenkombinationen eindeutig voneinander unterscheiden, zum Beispiel <SHIFT ALT ENTER> von <SHIFT RETURN>:

<SHIFT ALT ENTER > ist so abfragbar: PEEK(211) : 1+8 = 9

PEEK(212) : 77-1 = 76

<SHIFT RETURN> sieht dagegen so aus: PEEK(211) : 1 = 1

PEEK(212) : 2-1 = 1

Ein kleines Programmbeispiel hierzu:

```
10 REM* Erkennung aller Tasten**
20 PRINT ''BITTE 'SHIFT-ALT-ENTER' DRUECKEN''
30 DO
40 : SH = PEEK(211) : TA = PEEK(212) :REM TASTE
                                       BERECHNEN
50 LOOP UNTIL (SH=9 AND TA=76)      :REM KOMBINATION
                                      GEDRUECKT?
60 PRINT ''DANKE''
```

(H. Stöcklein/ef)

4. Redefinition der Funktionstasten

Wie in unserem Magazin bereits erwähnt, lassen sich die Funktionstasten mit folgender Zeile so belegen, daß sie wie im C 64-Modus von Programmen aus abfragbar sind:

```
FOR I=1 TO 8 : KEY I, CHR$(132+I) : NEXT
```

Will man nach dieser Eingabe aber mit der Editierung des Programms fortfahren, so störte bisher, daß die beim Einschalten vorgenommene Belegung der Funktionstasten (LIST, RUN und so weiter) nun nicht mehr vorhanden war. Hier kann mit einem einfachen POKE-Befehl Abhilfe geschaffen werden:

```
POKE 2564,0
```

Nun wird die ursprüngliche Belegung der Funktionstasten durch BANK 15:SYS 49152 oder <STOP-RESTORE> ohne Programmverlust wiederhergestellt. Mit SYS 49152 ist das sogar im Programmodus möglich (z.B. am Programmende).

Zur Funktion dieses »POKE«-Befehls: Die Speicherstelle 2564 ($0A04) dient dem Betriebssystem als Merkmal, ob die Tastatur-Code-Tabellen und die Funktionstastenbelegungen schon kopiert worden sind. Nach einem RESET wird dieses Flag auf $80 gesetzt, so daß bei einem Editorreset (Routine CINT, Einsprung bei $C000, beziehungsweise nach STOP-RESTORE) ein erneutes Kopieren erspart werden kann. Setzt man dieses Flag einfach von Hand wieder auf Null, werden beim nächsten Editorreset die Tastaturbelegungen wieder auf die im ROM festgelegten Werte zurückgesetzt. (H. Stöcklein/ef)

5. Künstlicher Fehler

Der Trick besteht darin, in die Basic-Routine, die bei einem Fehler aufgerufen wird, einzusteigen und die nötigen Parameter (Fehlernummer) mit zu übergeben. Da bei dieser Methode auch Fehlernummern größer als 43 vorkommen können, kann man hier leicht mit Hilfe der TRAP-Anweisung verschiedene Unterprogramme ersetzen oder auch spezielle Fehlermeldungen provozieren (z.B. »Falsche Eingabe!«). Aufruf:

```
SYS 19775,, Fehlernummer
```

(2 Kommata sind Pflicht, Fehlernummer > 125 entspricht keinem Fehler). Listing 2 ist ein kurzes Demoprogramm. (K. H. Guckler/ef)

```
10 BANK 1 : PRINT "OK"
20 PRINT "+++ FEHLER = 66"
30 TRAP 100
40 :
50 SYS 19775,,66
60 :
70 PRINT "{DOWN}AN DER FEHLERROUTINE VORBEI"
80 END
90 :
100 IF ER=66 THEN PRINT "{DOWN}FEHLER = 66 !!!"
110 IF ER<>66 THEN PRINT "FEHLER = "ER,,ERR$(ER)
120 GET KEY A$
130 RESUME NEXT
```

Listing 2. Beispiel für die künstliche Fehlermeldung

6. INPUT mit Punkt und Komma

Durch einen einfachen Trick kann man beim INPUT-Befehl auch die Zeichen ».« und »,« eingeben.

Drei POKE-Befehle sorgen für eine einfachere Eingabe. Zuerst wird der Tastaturpuffer auf zwei Zeichen gesetzt:

```
POKE 208, 2:REM (Zwei Zeichen im Tastaturpuffer)
```

Der nächste POKE-Befehl erlaubt dann die Eingabe von Punkt und Komma:

```
POKE 842,34:REM (Anführungszeichen)
```

Um den Schönheitsfehler (stehengebliebenes Anführungszeichen) zu beseitigen, löscht man einfach das letzte Zeichen:

```
POKE 843,20:REM (letztes Zeichen löschen)
```

Der INPUT-Befehl muß unmittelbar nach den POKE-Befehlen stehen. Einer Eingabe von ».« und »,« steht nun nichts mehr im Wege. (M. Völker/ef)

7. PEEKs und POKEs

Nach LIST nur die Zeilennummern sichtbar:
`POKE 774,38:POKE 775,160`

Nach LIST Reset:
`POKE 774,61:POKE 775,255`

<RUN/STOP> und <RUN/STOP RESTORE> ausschalten:
`POKE 808,PEEK(808)-3`

<RUN/STOP RESTORE> ausschalten:
`POKE 792,51:POKE 793, 255`

Nach SAVE Reset:
`POKE 818,61:POKE 819,255`

BOOT-Routine aufrufen:
`SYS 65366 oder SYS 65363,0,8`
(die erste Zahl ist die Laufwerksnummer, die zweite die Geräteadresse)

Zwischen 40- und 80-Zeichen-Modus umschalten:
`SYS 49194`

Software-Reset:
`BANK 15:SYS 4352`

Direkt in den C 64-Modus springen:
`SYS 65357 oder SYS 57931`

Nach einem Reset in den C 64-Modus springen oder dort bleiben:

```
MONITOR
M 1FFF5 1FFF9
```

die ersten Zahlen ändern in 43 42 4D 4D FF 05 FF 3D
danach X eingeben

Hard- oder Software-Reset geben.

Sie sind nun im C 64-Modus und können ihn auch durch Reset nicht wieder verlassen. (P. Vandekerckhove/ef)

8. PEEK-POKE-SYS, Nummer 2

Wiederherstellen eines Basic-Programms nach NEW oder einem Reset (OLD, RENEW):
`POKE 1+256*PEEK(46),1:RENUMBER`

Neuladen des VDC-Original-Zeichensatzes:
`SYS 65378`

Banknummer setzen oder lesen:
`POKE 981, banknummer`
oder
`BN=PEEK(981)`

Aktuelle Basic-Zeilennummer:
`PEEK (59)+256*PEEK(60)`

Aktuelle Zeilennummer für DATA:
`PEEK (65)+256*PEEK(66)`

Tastaturpuffer:
`POKE 842,x bis POKE 851,x`

Anzahl der Zeichen im Tastaturpuffer:
`PEEK (208)`

Der Bereich des Video-Chips VIC ist ohne weiteres nicht durch POKE ansprechbar. Besonders störend ist dies bei der Sprite-Steuerung. Die entsprechenden Register sind jedoch in einen zugänglichen Bereich gespiegelt. Die X- und Y-Koordinaten der Sprites 1 bis 8 finden Sie ab Adresse 4566 ($11D6) bis Adresse 4581 ($11E5). Die entsprechenden höchstwertigen Bits stehen in Speicherstelle 4582 ($11E6). (K. Guckler/ef)

9. POKEs, PEEKs & SYS

Als erstes nehmen wir uns mal die aus Low- und High-Byte bestehenden Vektoren der Restore- und der Reset-Taste vor. Theorie dazu (Fortgeschrittene können das überlesen): Vektoren sind die Zeiger, die auf bestimmte Maschinensprache-Routinen gerichtet sind. High- und Low-Bytewerte einer Zahl errechnen wir wie folgt: Beispielzahl: 4846 dezimal. Diese Zahl wandeln wir in hexadezimale Schreibweise mit PRINT HEX$(4864) um und erhalten $1300. Diese trennen wir bei der Mitte in »13« und »00« auf. Diese Zahlen wandeln wir wieder in dezimale Schreibweise durch PRINT DEC("13"), DEC("00") um und erhalten dann die Zahlenpaare dezimal 19 und dezimal 0. Die rechts ausgegebene Zahl ist immer das Low-Byte, die andere das High-Byte.
1. Praktische Anwendung: In Bank 15 unter der dezimalen Adresse 792 und 793 liegt der NMI-Vektor der Restore-Taste. Diesen Vektor können Sie nun beliebig verbiegen. Beispiele: POKE 792,0: POKE 793,64 = bei Restore Einschaltbild; POKE792,75: POKE793,226 = bei Restore C 64-Modus; POKE 792,0: POKE 793,224 = Reset bei Restore.
2. Praktische Anwendung: In Bank 1 an den (dezimalen) Adressen 65528 und 65529 liegt der Reset-Vektor. Auch diesen können Sie beliebig manipulieren.

Beispiele: BANK 1: POKE 65528,75: POKE 65529,266 = bei Reset 64'er-Modus; BANK 1: POKE 65528,33 = bei Reset Bildschirmchaos.

Denken Sie daran, daß Sie eigene Maschinensprache-Programme genauso gezielt durch Druck von Reset oder Restore aufrufen können, wie auch ROM-Routinen.

Nun folgen diverse Einzeltricks, mit denen Sie Ihre Programme verbessern.

1. POKE 820, 189 macht Escape im Quote-(Anführungszeichen-) Modus sichtbar.
2. POKE 808, 225 läßt den Computer total abstürzen.
3. POKE 0, PEEK (0) OR 64: POKE 1,PEEK (1) AND 191 schaltet auf DIN-Modus um (wer bisher nur POKE 0, PEEK (0) OR 64: POKE 1,0 benutzt hat, dürfte festgestellt haben, daß das im 40-Zeichenmodus nur mit einer Farbe funktioniert).
4. POKE 0, 47 schaltet wieder zurück in den ASCII-Modus.
5. POKE 2757, 129. Im DIN-Modus werden alle Tastatureingaben nur mit Großschrift wiedergegeben; Zahlen bleiben jedoch erhalten.
6. POKE 2757, 128 sperrt im 80-Zeichen-Modus die Umschaltung von ASCII auf DIN. Dieser POKE hat aber noch eine Funktion: Wird nach der Eingabe im Direktmodus von ASCII auf DIN umgeschaltet und werden dann Zeichen eingegeben, so erscheinen die Buchstaben A bis Z geSHIFTet, der Rest ungeSHIFTet.
7. PEEK (235) gibt die aktuelle Zeile aus (evtl. für Textverarbeitungen).
9. PEEK (236) gibt die aktuelle Spalte aus.
10. POKE 902, 128 gibt bei allen Befehlen Syntax-Error aus.
11. POKE 900, 20: kein Befehl hat Wirkung.
12. SYS 52684, 255 bewirkt bei 80-Zeichen extrem große Schrift. Durch den Interlace-Modus flimmert es stark.
13. SYS 65366 ruft die »BOOT CALL ROUTINE« auf.
14. POKE 244, 0 schaltet im Programm-/RUN-Modus den Quotemodus aus.
15. Ist PEEK (204) gleich 128, dann ist der ASCII-Zeichensatz aktiv.
16. POKE 245, 64 sperrt Groß/Klein-Umschaltung.
17. POKE 245, 0 gibt Groß/Klein-Umschaltung frei.
18. POKE 248, 128: Bildschirmrollen aus.
19. POKE 248, 0: Bildschirmrollen an.
20. SYS 51602 gibt Ton aus (= chr$(7)).
21. SYS 57344: Reset.
22. SYS 16384: Reset; Attribute wie Farbe werden beibehalten.
23. POKE 2595, 2 vergrößert den Tastenpuffer um 20 Elemente.
24. POKE 2595, 0 schaltet den Tastenpuffer aus.
25. POKE 53296, 255: Fastmodus für 64'er-Modus.
26. POKE 53296, 0 Fastmodus im 64'er-Modus aus.
27. Wenn PEEK (215) = 128 ist, dann ist der 80-Zeichen-Modus aktiv.
28. POKE 43, 1: POKE 44, 28: geben Sie diese POKEs im C 64-Modus ein. Anschließend wird das sich vorher im C 128-Modus befindende Programm im C 64-Modus zur Bearbeitung freigegeben.
29. POKE (PEEK (45) + 256 * PEEK (46) + 1), 28: SYS DEC ("4F4F") bringt das durch Reset oder NEW gelöschte Programm wieder zurück (= OLD).
30. POKE (PEEK (45) + 256 * PEEK (46)), 1: SYS 20303: SYS 20354 gleiche, jedoch sicherere Funktion als unter Punkt 29.
31. POKE 2593, 1: ersetzt GETKEY.
32. POKE 243, 1: revers an.
33. POKE 243, 0: revers aus.
34. In PEEK (6212) steht die Tastaturnummer der soeben durch GETKEY abgefragten Taste. Damit werden zum Beispiel die »1« über dem Q und die »1« vorm Zehnerblock getrennt abgefragt.
35. POKE 208, 0: WAIT 208, 1: wie Punkt 31.
36. POKE 808, 112: RUN/STOP aus.
37. POKE 808, 110: RUN/STOP ein.
38. POKE 208, 0: löscht Tastenpuffer.
39. SYS 65520, zeile, spalte ersetzt CHAR.
40. POKE 53265, 27: FAST im 40-Zeichenmodus.
41. POKE 53265, 11: SLOW im 40-Zeichenmodus.
42. POKE 241, PEEK (241) OR 214: PRINT "TEXT": Der TEXT wird blinkend dargestellt.
43. POKE 241, PEEK (241) OR 215: PRINT "TEXT": Der TEXT wird unterstrichen dargestellt.
44. POKE 241, PEEK (241) OR 216: PRINT "TEXT": Der TEXT wird invertiert dargestellt.
45. POKE 241, PEEK (241) OR 217: PRINT "TEXT". Der TEXT wird im DIN-Zeichensatz dargestellt.
46. Die Punkte 42 bis 45 können auch kombiniert werden. Beispiel: POKE 241, PEEK (241) OR 214 + 215: PRINT "TEXT". Der Text wird unterstrichen und blinkend dargestellt.
47. SYS 57931: GO64 ohne Kontrollfrage.
48. POKE 981, 128: jeder darauf folgende SYS-Befehl wird falsch adressiert.
49. POKE 249, 128 verhindert den Ton durch <Control G>.
50. SYS 19910 entspricht einem END, jedoch ohne »READY.«-Meldung. (A. Glootz/ef)

10. Interlace-Basic

Im 64'er-Sonderheft 22 über den C 128 stellten wir Ihnen eine Routine vor, die den Interlace-Modus des VDC aktiviert. Heute präsentieren wir das »TNT-Basic« (Listing 3), das die-

se Fähigkeit Ihres C 128 besser unterstützt und noch eine Menge weiterer Befehle bietet. So erhalten Sie durch diese Erweiterung nicht nur etliche Befehle und Funktionen, die die Programmierung des VDC unterstützen, Sie finden auch viele Anweisungen, die dem C 128 bisher einfach fehlten: OLD macht ein versehentlich eingegebenes NEW rückgängig, FIND findet Basic-Befehle und String-Konstanten oder eine Anweisung wie LOCKS, die die Tastenkombination <RUN/STOP RESTORE> blockiert. Syntax und Bedeutung der neuen Anweisungen und Funktionen entnehmen Sie bitte der Tabelle 1. Sie starten TNT-Basic durch SYS 4864.

(M. Ilse/ef)

```
                    BEFEHLS-ÜBERSICHT
                     FÜR TNT-BASIC

RESET FÜHRT EINEN RESET AUS
MERGE LÄDT EIN PROGRAMM HINTER DAS IM SPEICHER STEHENDE PROGRAMM
MERGE2 VERBINDET ZWEI PROGRAMME NACH EINEM MERGE-BEFEHL
OLD STELLT EIN BASIC-PROGRAMM NACH NEW WIEDER HER
OLDKEY ORIGINAL-FUNKTIONSTASTENBELEGUNG WIEDERHERSTELLEN
FIND XYZ LISTET ALLE ZEILEN,IN DENEN DER SUCHBEGRIFF HINTER FIND
VORKOMMT. WERDEN TEXTE IN ANFÜHRUNGSSTRICHEN GESUCHT, MÜSSEN
DIESE AUCH IN ANFÜHRUNGSZEICHEN EINGEGEBEN WERDEN.
CUON SCHALTET DEN CURSOR AN
COFF SCHALTET DEN CURSOR AUS
LOCKS BLOCKIERT DIE STOP-TASTE UND <RUN/STOP RESTORE>
UNLOCKS GIBT DIE BLOCKIERTE STOP-TASTE FREI
LLIST X-Y BASIC-PROGRAMM AUF DRUCKER (GERÄTEADRESSE 4,
SEKUNDÄRADRESSE 7)
LSET MIT "LSET GERÄTEADRESSE,SEKUNDÄRADRESSE" KÖNNEN GERÄ-
TEADRESSE UND SEKUNDÄRADRESSE FÜR LLIST NEU GESETZT WERDEN
OFF SCHALTET DIE BASIC-ERWEITERUNG AB UND SETZT DEN ALTEN RESET-
VEKTOR (KEIN NEUSTART VON TNT-BASIC)
OFFH SCHALTET DIE BASICERWEITERUNG AB. ALLERDINGS BLEIBT DER RE-
SET-VEKTOR ERHALTEN, EIN "LOCKS" WIRD DURCHGEFÜHRT UND BEI
BETÄTIGUNG DER RESTORE-TASTE WIRD DIE ERWEITERUNG WIEDER
GESTARTET
SCREEN X,Y POKE FÜR DAS VDC-RAM
INSCR (X) EIN PEEK FÜR DAS VDC-RAM
XKEY X,"" WIE KEY, NUR IST DIE NUMMER JEWEILS UM EINS KLEINER UND
ES KÖNNEN ALLE 10 TASTEN BELEGT WERDEN
CGOTO X BERECHNETES GOTO
WVREG X,Y SCHREIBEN EINES WERTES IN EIN VDC-REGISTER
CGOSUB X BERECHNETES GOSUB
LWIND OHNE PARAMETER WIRD DAS FENSTER = MAXIMALE BILDSCHIRMGRÖSSE
LWIND X MIT NUR EINER ZEILE: BILDSCHRIMFENSTER = ZEILE
LWIND X,Y BILDSCHIRMFENSTER = ERSTE BIS LETZTE ZEILE
DPOKE X,Y ZUM POKEN EINES 16 BIT-INTEGER-WERTES INS RAM
DPEEK (X) LIEFERT DEN WERT DER ADRESSE X+256*WERT DER ADRESSE X+1
OFINPUT A INPUT OHNE AUSGABE EINES FRAGEZEICHENS
TKEY WARTET AUF DAS DRÜCKEN EINER TASTE, DER TASTATURPUFFER WIRD
GELÖSCHT
INTERLACE OHNE PARAMETER WIRD DER INTERLACE-MODUS DES VDC AKTI-
VIERT, UND DER OBERE AUF DEN UNTEREN BILDSCHIRM KOPIERT
INTERLACE1 KOPIERT DEN OBEREN AUF DEN UNTEREN BILDSCHIRM
INTERLACE2 KOPIERT DEN UNTEREN AUF DEN OBEREN BILDSCHIRM
INTERLACE3 VERTAUSCHT DEN UNTEREN UND DEN OBEREN BILDSCHIRM HIER-
BEI WIRD DER ZEICHENSPEICHER DES VDC FÜR GROSS/GRAPHIK-SCHRIFT
BENUTZT, UND DESHALB WIRD DER ZEICHENSATZ AUS DEM ROM NOCHEINMAL
IN DAS RAM DES VDC-KOPIERT.
INTERLACE4 WIE INTERLACE3, NUR WIRD DER ZEICHENSATZ NICHT KOPIERT
INTERLACEOFF SCHALTET DEN INTERLACE-MODUS AB. MAN KANN IN EINEM
PROGRAMM EIN ECHTES WINDOW SIMULIEREN, INDEM MAN AM ANFANG DES
PROGRAMMS "INTERLACE:INTERLACEOFF" EINGIBT,DEN ZU RETTEN
BILDSCHIRM MIT INTERLACE1 IM PROGRAMM RETTET UND MIT INTERLACE2
SPÄTER ZURÜCKHOLT.

          Sondertasten
          ALT                   = INTERLACE4
          ALT+SHIFT             = INTERLACE3
          ALT+CONTROL           = INTERLACE2
          ALT+COMMODORE         = INTERLACE1
          ALT+SHIFT+COMMODORE   = INTERLACE
          ALT+SHIFT+CONTROL     = INTERLACEOFF

CLIST MODIFIZIERT DIE LIST-ROUTINE, BEIM LISTEN WERDEN ALLE
BEFEHLE DER BASIC-ERWEITERUNG UND ALLE REM-ZEILEN INVERS
AUSGEGEBEN. AUSSERDEM WIRD BEIM BEFEHL WKEY AUF EINEN TASTENDRUCK
GEWARTET. DIE FORMATZERSTÖRENDEN ZEICHEN CHR$(13), CHR$(141) UND
CHR$(27) WERDEN ALS CHR$-KODE IN DEN STRING EINGEFÜGT. EIN
ERNEUTES CLIST HEBT DIE MODIFIZIERUNG AUF
USED (X) GIBT DIE LÄNGE DES BASIC-PROGRAMMS IN BYTE AN
RVREG (X) GIBT DEN INHALT DES VDC-REGISTERS X AN
LOW (X) LIEFERT DAS LOW-BYTE DES WERTES X
HIGH (X) LIEFERT DAS HIGH-BYTE DES WERTES X
RBANK (X) LIEFERT DIE NUMMER DER EINGESCHALTETEN BANK
```

Tabelle 1. Befehlserklärung und Syntax des »TNT-Basic«

```
Name : tnt-basic          1300 1c01
-----------------------------------
1300 : a9 ff 85 fe a9 f8 85 fd  5e
1308 : a9 fd 8d b9 02 a0 01 a9  c7
1310 : 17 a2 7f 20 af 02 88 a2  cf
1318 : 7f a9 d7 20 af 02 20 24  3a
1320 : ca a0 00 8c 00 ff b9 6b  8a
1328 : 13 c9 40 d0 0e 24 d7 10  cb
1330 : 0d a2 13 a9 20 20 69 92  56
1338 : ca d0 f8 20 69 92 c8 c0  7c
1340 : 84 d0 e3 a2 05 bd 65 13  73
1348 : 9d 0c 03 ca 10 f7 a9 5f  2c
1350 : a2 14 8d fc 02 8e fd 02  90
1358 : a9 af 8d 00 0a a9 1b 8d  b2
1360 : 01 0a 4c b5 1b ef 13 15  d8
1368 : 14 50 14 0d 0b 0e 05 93  a7
1370 : 3e 20 20 40 d4 ce d4 2d  40
1378 : c2 41 53 49 43 20 28 d3  56
1380 : 4f 46 54 2d c2 41 53 49  c3
1388 : 43 2d c5 52 57 45 49 54  8b
1390 : 45 52 55 4e 47 29 40 20  1c
1398 : 20 3c 3e 20 20 40 3c 43  e5
13a0 : 3e 20 42 59 20 d4 ce d4  38
13a8 : 20 2d 20 53 4f 46 54 57  f8
13b0 : 4f 52 4b 20 28 cd 41 52  9a
13b8 : 54 49 4e 20 c9 4c 53 45  1f
13c0 : 29 40 20 3c 3e 20 20 40  7f
13c8 : 31 32 32 33 36 35 20 c2  18
13d0 : 41 53 49 43 20 c2 59 54  9c
13d8 : 45 53 20 46 52 45 45 20  3c
13e0 : 30 31 2e 31 31 2e 31 39  16
13e8 : 38 37 40 20 20 3c 0d 85  f3
13f0 : fc a9 14 a0 6e 20 e2 43  d4
13f8 : 90 0e a2 00 29 7f 09 40  6b
1400 : 4c b2 43 a5 fc 4c 26 43  7c
1408 : a9 14 a0 ec 20 e2 43 90  c8
1410 : f2 a2 ff d0 e7 20 1b 14  81
1418 : 4c 2e 51 29 3f 09 80 e8  05
1420 : f0 27 aa a9 14 a0 6e 85  8f
1428 : 25 84 24 a0 00 ca 10 0f  61
1430 : b1 24 48 e6 24 d0 02 e6  81
1438 : 25 68 10 f4 30 ef c8 b1  3d
1440 : 24 30 05 20 0c 56 d0 f6  66
1448 : 60 aa a0 ec a9 14 d0 d7  f1
1450 : 29 3f 0a a8 b9 0d 15 48  99
1458 : b9 0c 15 48 4c 80 03 29  8d
1460 : 3f 0a a8 b9 49 15 48 b9  d7
1468 : 48 15 48 4c 56 79 52 45  db
1470 : 53 45 d4 4d 45 52 47 45  d3
1478 : b2 4d 45 52 47 c5 4f 4c  e5
1480 : c4 46 49 4e c4 43 55 4f  de
1488 : ce 43 4f 46 c6 4c 4f 43  27
1490 : 4b d3 55 4e 4c 4f 43 4b  c7
1498 : d3 4c 4c 49 53 d4 4f 46  73
14a0 : c6 53 43 52 45 45 ce 58  95
14a8 : 4b 45 d9 43 47 4f 54 cf  54
14b0 : 57 56 52 45 c7 43 47 4f  c2
14b8 : 53 55 c2 4c 57 49 4e c4  72
14c0 : 44 50 4f 4b c5 4f 46 49  ec
14c8 : 4e 50 55 d4 54 4b 45 d9  97
14d0 : 49 4e 54 45 52 4c 41 43  11
14d8 : c5 43 4c 49 53 d4 57 4b  4b
14e0 : 45 d9 4c 53 45 d4 45 52  44
14e8 : 41 53 c5 00 49 4e 53 43  1f
14f0 : d2 55 53 45 c4 52 56 52  c7
14f8 : 45 c7 4c 4f d7 48 49 47  91
1500 : c8 52 42 41 4e cb 44 50  9f
1508 : 45 45 cb 00 3c ff 56 15  2a
1510 : 66 15 88 15 f6 16 85 17  2a
1518 : 8b 17 91 17 97 17 aa 17  81
1520 : 01 18 aa 15 c9 15 e4 16  80
1528 : d7 18 ea 16 f8 18 89 18  2f
1530 : b8 18 37 19 44 19 bf 1a  26
1538 : 37 19 9d 1b c1 1b 00 d4  65
1540 : ce d4 2d d3 4f 46 54 00  b7
1548 : 7d 16 c1 16 50 16 06 16  fd
1550 : 0e 16 16 16 1e 16 00 a9  98
1558 : 00 f0 0b 85 2e a9 00 85  7f
1560 : 2d a9 00 8d 58 15 60 ad  1f
1568 : 58 15 d0 0a a5 2d 8d 5e  77
1570 : 15 a5 2e 8d 58 15 ad 10  9a
1578 : 12 ac 11 12 38 e9 02 b0  a3
1580 : 01 88 85 2d 84 2e 4c 2c  10
1588 : 91 f0 18 c9 f9 f0 03 4c  9c
1590 : 6c 79 20 45 a8 a2 4c bd  b6
1598 : a8 ce 9d 00 10 ca 10 f7  96
15a0 : 4c 80 03 a0 01 98 91 2d  77
15a8 : 4c e5 5e 20 03 88 86 fc  0b
15b0 : a5 17 c9 40 b0 11 a2 12  9d
15b8 : 20 21 1a a5 16 a2 13 20  a7
15c0 : 21 1a a5 fc 4c 1f 1a 4c  b6
15c8 : 28 7d f0 08 20 f4 87 e0  75
15d0 : 0a 4c e7 60 a2 ff a0 00  b3
15d8 : e8 a9 08 85 77 bd 00 10  cd
15e0 : f0 19 86 fc 85 78 a2 06  51
15e8 : bd 00 16 ca d0 02 05 fc  af
15f0 : 20 69 92 8a 10 f2 20 3f  52
15f8 : 61 a6 fc e0 09 d0 d9 60  47
1600 : 2c 30 20 59 45 4b 58 20  c8
1608 : 64 16 a5 16 4c 96 16 20  b6
1610 : 64 16 a5 17 4c 96 16 ac  f7
1618 : d5 03 a9 00 4c d9 16 20  05
1620 : 64 16 a5 16 a6 17 85 fc  ef
1628 : 86 fd 68 85 16 68 85 17  61
1630 : ae d5 03 a0 00 a9 fc 20  1f
1638 : 74 ff 85 fb e6 fc d0 02  2a
1640 : e6 fd ae d5 03 a0 00 a9  14
```

Listing 3. »TNT-Basic« unterstützt VDC und Basic

```
1648 : fc 20 74 ff a4 fb 4c d9  80
1650 : 16 20 64 16 a5 17 d0 23  ef
1658 : a6 16 e0 25 b0 1d 20 5a  0f
1660 : 1a 4c 96 16 68 aa 68 a8  d8
1668 : a5 17 48 a5 16 48 98 48  f6
1670 : 8a 48 20 da 77 20 15 88  60
1678 : 4c 45 a8 4c 28 7d 20 64  d2
1680 : 16 a5 17 c9 40 b0 f4 c6  53
1688 : 16 a6 16 e8 d0 02 c6 17  fa
1690 : 20 ae 16 20 ae 16 a8 68  a0
1698 : 85 16 68 85 17 c0 00 f0  4c
16a0 : 03 4c d4 84 a2 04 94 63  f2
16a8 : ca 10 fb 68 68 60 a2 13  c0
16b0 : a5 16 20 21 1a a2 12 a5  d7
16b8 : 17 20 21 1a 20 58 1a 4c  31
16c0 : 58 1a ad 10 12 ae 11 12  92
16c8 : 38 e5 2d b0 02 ca 38 e9  7f
16d0 : 02 b0 01 ca a8 8a 38 e5  4f
16d8 : 2e c0 00 d0 04 c9 00 f0  f1
16e0 : c3 18 4c c9 84 20 f3 18  45
16e8 : 4c e2 59 20 1d 5a 20 86  32
16f0 : 03 20 e5 16 4c f6 4a 20  25
16f8 : 83 17 f0 4f a2 00 86 fb  69
1700 : c9 22 d0 11 e6 fb 20 7d  fa
1708 : 17 c9 22 d0 08 a4 fb f0  1e
1710 : 04 20 80 03 98 9d 00 0b  31
1718 : e8 c9 00 d0 e9 a5 2e a6  d1
1720 : 2d a0 01 d0 1d 20 98 55  d8
1728 : a0 02 20 ec 42 aa c8 20  4c
1730 : ec 42 20 25 51 20 b5 4b  6e
1738 : a0 00 20 ec 42 aa c8 20  5b
1740 : ec 42 85 62 86 61 20 ec  c9
1748 : 42 d0 12 60 c8 e8 bd 00  4e
1750 : 0b f0 d2 20 ec 42 dd 00  e4
1758 : 0b f0 f1 d0 1c 88 84 fa  80
1760 : a0 04 a2 00 20 ec 42 f0  ff
1768 : cf c9 22 d0 06 a9 01 45  fb
1770 : fa 85 fa a5 fb c5 fa f0  5c
1778 : da c8 d0 e6 60 e6 3d d0  9b
1780 : 02 e6 3e 4c c9 03 20 45  ce
1788 : a8 4c 6f cd 20 45 a8 4c  53
1790 : 9f cd a2 17 a9 a2 d0 04  9c
1798 : a9 6e a2 f6 8e 29 03 8d  59
17a0 : 28 03 85 fc a9 01 08 a5  59
17a8 : fc 28 60 20 45 a8 20 bd  6a
17b0 : ff a9 00 a2 04 a0 07 20  7a
17b8 : ba ff 20 c0 ff a2 00 20  e7
17c0 : c9 ff 20 86 03 20 e2 50  bf
17c8 : 20 98 55 a9 00 20 c3 ff  cf
17d0 : a9 00 85 ba 4c c9 ff a2  8a
17d8 : ff 78 9a d8 20 42 e2 20  b5
17e0 : 09 e1 20 93 e0 20 56 e0  7e
17e8 : 20 00 c0 20 7a 41 20 51  11
17f0 : 42 20 45 40 20 00 13 ad  45
17f8 : 04 0a 09 01 8d 04 0a 4c  1e
1800 : 3c 40 08 20 80 03 a2 05  17
1808 : bd 84 18 9d 0c 03 ca 10  e5
1810 : f7 a9 40 8d 01 0a a9 03  ab
1818 : 8d 00 0a a9 78 8d fc 02  49
1820 : a9 4c 8d fd 02 20 cb 1a  97
1828 : 78 a9 fa 8d 15 03 a9 65  c0
1830 : 8d 14 03 58 28 f0 0f 78  ca
1838 : a9 18 8d 19 03 a9 67 8d  aa
1840 : 18 03 58 4c 92 17 20 98  0d
1848 : 17 a9 ff 85 fe a9 f8 85  10
1850 : fd a9 fd 8d b9 02 a0 01  83
1858 : a9 e2 a2 7f 20 af 02 88  a3
1860 : a2 7f a9 24 4c af 02 d8  ad
1868 : 20 3d f6 c9 ff d0 12 20  2d
1870 : 43 13 78 a9 40 8d 18 03  67
1878 : a9 fa 8d 19 03 58 20 98  ca
1880 : 17 4c 40 fa 21 43 cd 51  33
1888 : a9 48 20 f3 18 84 fc 85  80
1890 : fd 20 f0 18 85 fb 98 a0  b8
1898 : 00 a2 fc 8e b9 02 ae d5  0c
18a0 : 03 20 77 ff a5 fb e6 fc  61
18a8 : d0 02 e6 fd a0 00 a2 fc  81
18b0 : 8e b9 02 ae d5 03 4c 77  07
18b8 : ff 48 a5 15 85 fc a9 01  d0
18c0 : 85 15 68 20 62 56 48 a5  33
18c8 : fc 85 15 86 fc 84 fd 20  c9
18d0 : 98 55 68 a6 fc a4 fd 60  af
18d8 : 20 f4 87 e0 25 b0 0e 86  8d
18e0 : fc 20 09 88 20 45 a8 8a  24
18e8 : a6 fc 4c 21 1a 4c 28 7d  e3
18f0 : 20 5c 79 20 d7 77 4c 15  35
18f8 : 88 d0 03 4c 24 ca 20 2f  aa
1900 : 19 e0 19 b0 30 86 fc 20  51
1908 : 86 03 f0 0b 20 2c 19 e0  37
1910 : 19 b0 22 e0 00 d0 02 a6  02
1918 : fc 8a e5 fc 90 17 a5 fc  c5
1920 : 85 e5 86 e4 20 50 c1 a9  b5
1928 : 00 4c 0c 56 20 5c 79 20  27
1930 : ee 91 4c f4 87 4c 28 7d  0f
1938 : a9 00 85 d0 a5 d0 f0 fc  fb
1940 : a9 00 85 d0 60 d0 75 a2  0c
1948 : 14 20 5a 1a 85 fc c9 10  ce
1950 : f0 06 20 de 6a 20 de 6a  1f
1958 : a0 05 b9 0e 1a be 14 1a  c7
1960 : 20 21 1a 88 10 f4 a9 10  18
1968 : 8d 2f 0a a5 fc c9 10 d0  c4
1970 : 01 60 a2 06 a0 04 a9 18  0c
1978 : 20 81 19 a2 02 a0 00 a9  6c
1980 : 08 8d ac 19 bd fe 19 85  e0
1988 : fc bd ff 19 85 fd b9 fe  b3
1990 : 19 85 fa b9 ff 19 85 fb  38
1998 : ce ac 19 20 a7 1a a9 00  f9
19a0 : a2 1e 20 21 1a e6 fb e6  14
19a8 : fd a5 fb c9 27 d0 ec 20  9d
19b0 : a7 1a a9 d0 a2 1e 4c 21  77
19b8 : 1a 4c 28 7d c9 fe f0 25  55
19c0 : 20 f4 87 e0 01 f0 ab e0  60
19c8 : 03 f0 5c e0 04 f0 58 e0  61
19d0 : 02 d0 e6 a2 04 a9 1f a0  94
19d8 : 06 20 81 19 a2 00 a0 02  23
19e0 : a9 0f 4c 81 19 20 80 03  ef
19e8 : c9 4a d0 11 20 80 03 a0  80
19f0 : 04 b9 1a 1a be 14 1a 20  d0
19f8 : 21 1a 88 10 f4 60 00 00  9d
1a00 : d0 07 00 10 d0 17 00 20  5c
1a08 : 00 28 21 20 13 12 4d 03  66
1a10 : 00 40 32 10 04 08 24 07  de
1a18 : 06 14 27 fc f5 20 19 a2  9c
1a20 : 1f 20 45 a8 4c cc cd 8a  2d
1a28 : 48 a2 08 a0 04 a9 18 20  06
1a30 : 81 19 a2 0a a0 00 a9 08  e8
1a38 : 20 81 19 20 d3 19 a2 02  f8
1a40 : a0 0a a9 30 20 81 19 a2  0d
1a48 : 06 a0 08 a9 28 20 81 19  91
1a50 : 68 c9 03 d0 a8 4c 0c ce  32
1a58 : a2 1f 20 45 a8 4c da cd  2f
1a60 : ad 7a 1a f0 06 ce 7a 1a  e4
1a68 : 4c 65 fa a5 d3 aa 29 08  21
1a70 : c9 00 f0 03 20 7b 1a 4c  b5
1a78 : 65 fa 00 e0 08 d0 04 a2  d3
1a80 : 04 d0 a4 e0 09 d0 04 a2  9e
1a88 : 03 d0 9c e0 0a d0 03 4c  02
1a90 : 72 19 e0 0c d0 03 4c d3  46
1a98 : 19 e0 0d d0 03 4c ef 19  03
1aa0 : e0 0b d0 1b 4c 47 19 a2  46
1aa8 : 18 20 5a 1a 09 80 20 21  02
1ab0 : 1a a0 03 b9 fa 00 be 0a  d1
1ab8 : 1a 20 21 1a 88 10 f4 60  0c
1ac0 : ae 07 03 a9 d5 a0 1a e0  74
1ac8 : 51 f0 03 a9 51 a8 8d 06  24
1ad0 : 03 8c 07 03 60 10 1e c9  ce
1ad8 : ff 10 1a 24 11 30 16 c9  69
1ae0 : 8f d0 07 a9 12 20 0c 56  cd
1ae8 : a9 8f c9 fe f0 58 c9 ce  42
1af0 : f0 67 4c 63 51 c9 0d f0  8d
1af8 : 0b c9 8d f0 07 c9 1b f0  76
1b00 : 03 4c 32 51 85 fc a5 4b  4d
1b08 : 48 a5 11 48 a5 55 48 86  a3
1b10 : fd 84 fe a9 22 20 0c 56  44
1b18 : a9 43 20 0c 56 a9 c8 20  03
1b20 : 0c 56 a9 28 20 0c 56 a9  d6
1b28 : 00 a6 fc 20 32 8e a9 29  4f
1b30 : 20 0c 56 a6 fd a4 fe 68  92
1b38 : 85 55 68 49 ff 85 11 68  ec
1b40 : 85 4b a9 22 d0 bb aa c8  41
1b48 : 20 ec 42 d0 03 4c 32 51  87
1b50 : 84 4b c9 27 b0 16 4c 9d  f9
1b58 : 51 aa c8 20 ec 42 d0 03  5f
1b60 : 4c 32 51 84 4b c9 0b b0  3b
1b68 : 07 4c b7 51 a2 00 f0 02  9f
1b70 : a2 ff 85 fc 86 fd 84 fe  7b
1b78 : a9 12 20 0c 56 a5 fc a6  88
1b80 : fd a4 fe 20 1b 14 29 7f  89
1b88 : 20 0c 56 a9 92 20 0c 56  80
1b90 : a5 fc c9 56 d0 03 20 38  07
1b98 : 19 a9 00 4c 2e 51 20 2f  5c
1ba0 : 19 86 fc 20 2c 19 8e b6  73
1ba8 : 17 a5 fc 8d b4 17 60 20  48
1bb0 : b5 1b 4c 03 40 78 a9 60  96
1bb8 : a2 1a 8d 14 03 8e 15 03  4c
1bc0 : 58 60 78 20 b3 77 a9 3f  87
1bc8 : 8d 00 ff a9 00 a8 a6 2d  c5
1bd0 : 86 fc a6 2e 86 fd 91 fc  dc
1bd8 : e6 fc d0 02 e6 fd a6 fd  a6
1be0 : ec 11 12 d0 f1 a6 fc ec  16
1be8 : 10 12 d0 ea a9 00 8d 00  64
1bf0 : ff 20 c4 77 58 4c d9 51  11
1bf8 : ff 00 ff 00 ff 00 ff 5f  b6
1c00 : 41 1d 1c 0a 00 41 4e b2  c1
```

Listing 3. (Schluß)

11. Input 128

Schon häufiger haben wir verbesserte INPUT-Routinen für den C 128 veröffentlicht. »Input 128« ist komplett in Assembler geschrieben, erlaubt eine einfache Parameterübergabe und besticht durch seine Geschwindigkeit. Input 128 ist nur im 80-Zeichen-Modus lauffähig. Nachdem Sie den DATA-Lader (Listing 4) eingetippt und gespeichert haben, wird er mit RUN gestartet. Nun wird die eigentliche Maschinensprache-Routine in den Speicher geschrieben, der Basic-Start auf dezimal 8192 gesetzt und ein NEW ausgeführt. Die unteren 8 KByte gelten jetzt als gemeinsamer (»common«) Bereich für alle Speicherbänke, was zur Folge hat, daß in Bank 0 ein KByte und in Bank 1 sieben KByte

weniger zur Verfügung stehen.
Die Syntax lautet:
SYS 4864,X,Y,L,,EZ$,VG$,IN$
Dabei bedeuten die Variablen

X	:	x-Position der Eingabemaske (0 bis 79)
Y	:	y-Position der Eingabemaske (0 bis 24)
L	:	Länge der Eingabe (1 bis 80)
EZ$	:	in der Eingabemaske erlaubte Zeichen
VG$	:	Vorgabe, die in der Eingabemaske auftauchen soll (maximal »L« Zeichen)
IN$	:	String-Variable für die Eingabe

Die Variablen-Namen sind selbstverständlich frei wählbar. Es ist aber zu beachten, daß EZ$, VG$ und IN$ nichtdimensionierte Variablen sein müssen. X, Y und L dürfen dagegen sowohl als Konstanten wie auch als Feldelememte angegeben werden. Weiter sollte EZ$ mindestens ein Zeichen enthalten, sonst erscheinen wirre Zeichen im Bildschirmspeicher — es wäre auch keine Eingabe erlaubt. Listing 5 demonstriert die Anwendung von Input 128.

(J. Bosmann/ef)

```
100 REM ***********************************           <IIA>
110 REM *   I N P U T   1 2 8   DATALADER   *         <IQB>
120 REM *                                   *         <I28>
130 REM * WRITTEN IN 1987 BY JOCHEN BOSMANN *         <IA9
140 REM *                                   *         <JIE
150 REM * ANSCHRIFT:  CRANACHWEG 2          *         <J9C
160 REM *             4440 RHEINE 1         *         <J2C
170 REM ***********************************           <JAD
180 :                                                 <6CP
190 SCNCLR : PRINT "LESE INITIALISIERUNGSDATEN...     <9V6
200 :                                                 <1C5
210 REM DATA'S DER INITIALISIERUNGROUTINE             <JAD
220 :                                                 <0C1>
230 BANK 15: FOR I=2816 TO 2837: READ D$: D=DEC(D
    $): POKE I,D: S=S+D: NEXT                         <ASV>
240 IF S<>2428 THEN PRINT "FEHLER IN ZEILE 260":
    END                                               <0HF>
250 :                                                 <3SF>
260 DATA AD,00,FF,48,A9,00,8D,00,FF,AD,06,D5,09,0
    6,8D,06,D5,68,8D,00,FF,60                         <MLC>
270 :                                                 <2SB>
280 PRINT "LESE INPUTDATEN...                         <NUI>
290 :                                                 <5SN>
300 REM DATA'S VON INPUT128                           <JIE>
310 :                                                 <0S3>
320 FOR I=4864 TO 5395 STEP 21: CK=0: FOR J=I TO
    I+20                                              <D3D>
330 READ DA$: POKE J,DEC(DA$): CK=CK+DEC(DA$): NE
    XT : READ CH                                      <37A>
340 IF CH<>CK THEN PRINT "ERROR IN ZEILE" PEEK(65
    )+256*PEEK(66): END                               <9ME>
350 NEXT                                              <TBG>
360 :                                                 <3CD>
370 DATA 85,FA,85,FC,86,FB,84,FE,18,65,FE,85,FD,C
    6,FD,A2,0E,A5,FA,C9,50, 3627                      <Q5R>
380 DATA B0,0C,A5,FB,C9,19,B0,06,A5,FD,C9,50,90,0
    3,4C,3F,4D,20,F1,14,A0, 2527                      <AJU>
390 DATA 02,B1,49,99,A9,00,88,10,F8,20,D4,14,20,F
    1,14,A0,02,B1,49,99,A6, 2262                      <T5J>
400 DATA 00,88,10,F8,20,D4,14,A2,17,A5,FE,C5,A6,9
    0,D5,A9,43,20,1E,C0,A9, 2647                      <79B>
410 DATA 00,8D,2B,0A,20,6F,CD,A9,12,85,63,A9,15,8
    5,64,A5,FA,85,FC,20,E8, 2448                      <00Q>
420 DATA 14,A4,FE,88,A9,20,91,63,20,D2,FF,88,10,F
    8,20,E8,14,A5,A6,F0,14, 2791                      <I4B>
430 DATA A0,00,20,DE,14,B1,A7,91,63,20,D4,14,20,D
    2,FF,C8,C4,A6,D0,EE,A9, 2960                      <VOE>
440 DATA 00,85,FF,20,E8,14,20,E4,FF,F0,FB,A2,00,D
    D,FA,14,F0,07,E8,E0,08, 3042                      <HUP>
450 DATA 90,F6,B0,12,8A,0A,AA,BD,02,15,8D,00,0B,B
    D,03,15,8D,01,0B,6C,00, 1740                      <E2N>
460 DATA 0B,A4,A9,88,20,DE,14,D1,AA,20,D4,14,F0,0
    6,88,10,F3,4C,CC,14,20, 2370                      <4RR>
470 DATA D2,FF,A4,FF,91,63,A6,FC,E4,FD,F0,88,E6,F
    C,E6,FF,4C,96,13,A4,FE, 4081                      <R7D>
480 DATA 88,B1,63,C9,20,D0,03,88,10,F7,C8,84,FE,2
    0,F1,14,20,D4,14,A5,FE, 2817                      <N60>
490 DATA 20,99,92,F0,11,A0,00,B1,63,20,DE,14,91,3
    5,20,D4,14,C8,C4,FE,D0, 2618                      <T4N>
500 DATA F1,20,DE,14,A0,00,A5,FE,91,49,B9,35,00,C
    8,91,49,C0,02,D0,F6,20, 2648                      <OB3>
510 DATA D4,14,4C,9F,CD,A5,FC,C5,FA,D0,03,4C,CC,1
    4,C6,FC,C6,FF,4C,96,13, 3195                      <JJ8>
520 DATA A5,FC,C5,FD,D0,03,4C,CC,14,E6,FC,E6,FF,4
    C,96,13,A6,FC,E4,FA,F0, 3726                      <01R>
530 DATA 7B,C6,FC,20,E8,14,A4,FF,B1,63,88,91,63,2
    0,D2,FF,C8,A9,20,91,63, 3074                      <62C>
540 DATA C8,C4,FE,D0,EE,20,D2,FF,91,63,C6,FF,4C,9
    6,13,A4,FE,88,B1,63,C9, 3566                      <N7A>
550 DATA 20,D0,4F,A5,FD,C5,FC,F0,49,85,FC,88,B1,6
    3,C8,91,63,84,65,48,20, 3077                      <JKH>
560 DATA E8,14,68,20,D2,FF,A4,65,88,C6,FC,C4,FF,D
    0,E7,A9,20,91,63,48,20, 3143                      <NFJ>
570 DATA E8,14,68,20,D2,FF,4C,96,13,A5,FA,85,FC,4
    C,92,13,A5,FA,85,FC,20, 2971                      <11I>
580 DATA E8,14,A9,20,A0,00,91,63,20,D2,FF,C8,C4,F
    E,D0,F6,4C,92,13,A9,07, 2875                      <2HT>
590 DATA 20,D2,FF,4C,96,13,08,48,A9,00,8D,00,FF,6
    8,28,60,08,48,A9,7F,8D, 2144                      <D09>
600 DATA 00,FF,68,28,60,A6,FB,A4,FC,18,20,F0,FF,6
    0,20,5C,79,20,AF,7A,4C, 2625                      <1O1>
610 DATA DE,14,0D,9D,1D,14,94,13,93,1B,E5,13,2B,1
    4,3B,14,4B,14,74,14,AD, 1596                      <0GC>
620 DATA 14,B4,14,63,13,00,00,00,00,00,00,00,00,0
    0,00,00,00,00,00,00,00,  338                      <PVU>
630 :                                                 <4SJ>
640 REM BASICSTART HOCHLEGEN UND PROGRAMM LOESCHE
    N                                                 <G20>
650 :                                                 <7SV>
660 PRINT "PROGRAMM EINGELESEN UND INITIALISIERT!
    "                                                 <J0K>
670 PRINT : PRINT "AUFRUF: SYS4864,X,Y,L,,EZ$,VG$
    ,IN$:IN$=IN$                                      <2GJ>
680 POKE 46,32: POKE 8192,0: POKE 48,32: CLR : SY
    S 2816: NEW                                       <HE0>
```

Listing 4. »Input 128« ist eine komfortable Eingabe-Routine

```
100 REM ***********************************           <IIA>
110 REM *  I N P U T 1 2 8   DEMONSTRATION  *         <IQB>
120 REM *                                   *         <I2B>
130 REM * WRITTEN IN 1987 BY JOCHEN BOSMANN *         <IA9>
140 REM ***********************************           <JIE>
150 :                                                 <0S3>
160 COLOR 5,6: COLOR 6,1: GRAPHIC 5,1: FAST           <B2E>
170 EZ$="": ZA$="0123456789.": DIM AD$(6)             <932>
180 FOR I=032 TO 090: EZ$=EZ$+CHR$(I): NEXT           <GT4>
190 FOR I=193 TO 218: EZ$=EZ$+CHR$(I): NEXT           <P50>
200 DO WHILE A$<>"←"                                  <9T6>
210 : SCNCLR : PRINT : PRINT TAB(32) CHR$(14) CHR
    $(2)"ADRESSVERWALTUNG": PRINT                     <1LR>
220 : PRINT TAB(15)"PROGRAMM ZUR DEMONSTRATION DE
    R NEUEN INPUT-ROUTINE": PRINT                     <2JL>
230 : PRINT : PRINT TAB(15)"NACHNAME:" SPC(21)"<"     <JNL>
240 : PRINT : PRINT TAB(15)"VORNAME :" SPC(21)"<"     <JGQ>
250 : PRINT : PRINT TAB(15)"STRASSE :" SPC(21)"<"     <BDO>
260 : PRINT : PRINT TAB(15)"WOHNORT :" SPC(21)"<"     <805>
270 : PRINT : PRINT TAB(15)"LAND{4SPACE}:" SPC(21
    )"<"                                              <0V4>
280 : PRINT : PRINT TAB(15)"GEBURTS.:" SPC(11)"<"     <IB1>
290 : VG$="": FOR I=1 TO 5: SYS 4864,25,4+2*I,20,
    ,EZ$,VG$,IN$: IN$=IN$: AD$(I)=IN$: NEXT           <90A>
300 : VG$="01.01.1987": SYS 4864,25,16,10,,ZA$,VG
    $,IN$: IN$=IN$: AD$(6)=IN$                        <QVK>
310 : SCNCLR : PRINT AD$(2)" "AD$(1)                  <GPJ>
320 : FOR I=3 TO 6: PRINT AD$(I): NEXT                <U1M>
330 : PRINT : PRINT : PRINT TAB(10)"TASTE!": GET
    KEY A$                                            <K9C>
340 LOOP                                              <R39>
350 SCNCLR                                            <9PJ>
```

Listing 5. Ein Demo zu Input 128

Futter für den C128

Wer kennt das nicht? Da versucht man verzweifelt, eine Lösung für ein spezielles Problem zu finden: Der Drucker gibt nur wirre Zeichen aus, die Bildschirmausgabe ist plötzlich anders als eingeplant... Aber da gibt es noch ja die Tips & Tricks, in denen vielleicht auch die Lösung Ihres Problems zu finden ist.

Wenn Sie solche Tips & Tricks kennen, zögern Sie nicht, uns diese zuzusenden. Helfen Sie den C128-Besitzern mit Ihren Soft- und Hardware-Kniffen, die Sie inzwischen entdeckt haben. Ganz gleich, ob es sich dabei um Einzeiler oder um kurze Routinen handelt:

Schicken Sie Ihren Brief an:

Markt & Technik
Redaktion Sonderhefte
Kennwort: Tips & Tricks
Hans-Pinsel-Straße 2
8013 Haar

12. BAS.MON mit zwei Bildschirmseiten

»BAS.MON« (Listing 6) erleichtert das Programmieren von Basic und Assembler. Wer ständig zwischen Basic und dem eingebauten Maschinensprache-Monitor des C 128 wechselt, kennt das zeitraubende Suchen von Programmteilen und Speicherzellen. Das Aufrufen des Monitors und das immer neue Auflisten des Basic-Programms entfallen nun, da mit zwei Bildschirmseiten im 80-Zeichen-Modus gearbeitet wird.

Starten Sie das Programm nach BLOAD"BAS.MON. SYS7136" ON B15 mit SYS 7136. Nach dem Erscheinen der Copyright-Meldung wartet BAS.MON auf einen Tastendruck. Der Bildschirm wird gelöscht und BAS.MON ist initialisiert. Durch Drücken der Funktionstaste <F1> wird auf die zweite Bildschirmseite und gleichzeitig in den Monitor gewechselt, und zwar ohne Registeranzeige. Ein zweiter Druck auf <F1> aktiviert wieder die erste Bildschirmseite und man arbeitet wieder mit dem Basic-Interpreter — ohne die Ausgabe von »READY«. Beide Bildschirminhalte bleiben natürlich voll erhalten. Gefällt Ihnen die Farbe nicht, ändern Sie direkt beim Abtippen die Adresse $1B0C. Und nun viel Spaß beim Programmieren!

(H. Krüger/ef)

```
Name : bas.mon. sys7136  1b00 1bf9
----------------------------------
1b00 : a9 01 8d 30 d0 20 42 c1  2e
1b08 : 20 50 c1 a9 8d 85 f1 a9  16
1b10 : 00 8d 72 1b 60 a2 0a a0  5b
1b18 : 14 18 20 f0 ff 20 7d ff  55
1b20 : c2 c1 d3 2e cd cf ce 2e  70
1b28 : 20 28 d6 c4 c3 29 20 20  f1
1b30 : 2d 20 20 28 43 29 20 31  db
1b38 : 39 38 37 20 42 59 20 c8  60
1b40 : 45 4e 52 59 20 cb 52 bd  91
1b48 : 47 45 52 00 a5 d4 c9 58  9f
1b50 : f0 fa 20 42 c1 4c c0 1b  c5
1b58 : a9 6a 85 fa a9 1b 85 fb  78
1b60 : a9 fa a2 01 a0 08 20 65  e5
1b68 : ff 60 93 53 d9 37 31 34  6b
1b70 : 38 0d 10 78 a9 80 a2 1b  a1
1b78 : 8d 3c 03 8e 3d 03 58 60  c4
1b80 : a2 09 dd dd c6 f0 16 ca  bc
1b88 : 10 f8 a6 d0 ec 20 0a b0  31
1b90 : 06 9d 4a 03 e8 86 d0 a9  b1
1b98 : 7f 8d 00 dc 60 e0 00 f0  68
1ba0 : 03 4c ca c6 ad 72 1b c9  c3
1ba8 : 10 f0 15 a2 0c a9 10 20  59
1bb0 : cc cd 8d 2e 0a 8d 72 1b  99
1bb8 : a9 13 20 d2 ff 4c 8e b0  4b
1bc0 : a2 0c a9 00 20 cc cd 8d  8d
1bc8 : 2e 0a 8d 72 1b a9 13 20  39
1bd0 : d2 ff 20 cc ff 20 7a 41  b1
1bd8 : 20 38 52 58 4c ba 4d ff  84
1be0 : 20 00 1b 20 58 1b 20 73  91
1be8 : 1b 4c 15 1b 20 00 1b 20  81
1bf0 : 58 1b 20 73 1b 4c 52 1b  e0
1bf8 : 54 fd 01 fd 81 fd 01 ff  57
```

Listing 6. »Bas.Mon.« – Programmieren mit zwei Bildschirmseiten. Im Monitor löschen Sie bitte den Bildschirm vor der ersten Eingabe.

13. Verbessertes RENUMBER für den C 128

Bei größeren Basic-Programmen besteht, zum Beispiel aus Gründen der Übersichtlichkeit, oftmals der Wunsch einer Neustrukturierung des Programms. Zwar befindet sich im Basic 7.0 des C 128 der Befehl RENUMBER, aber mit diesem ist eine Umstrukturierung eines Programms nicht möglich, da mit diesem Befehl keine Zeilenbereiche verschoben werden können.

Um so erfreuter war ich, als im Sonderheft 10 eine RENUMBER-Routine zu finden war, die eine Verschiebung von Zeilenbereichen gestattete. Darüber hinaus bestach das Programm durch Geschwindigkeit (etwa 20mal schneller als der originale RENUMBER-Befehl). Leider sind im besagten Programm ganz erhebliche Fehler. So werden die Befehle TRAP und RESTORE bei der Neunumerierung ignoriert. Beim Befehl ON ... GOTO wird in der folgenden Sprungtabelle nur der erste Wert geändert. Außerdem unterscheidet das Programm nicht zwischen Basic-Token und Grafikzeichen in Strings, was mitunter zu recht eigenartigen Veränderungen des Programms führt.

Eine weitere Besonderheit weist die RENUMBER-Routine auf, die der Verfasser aber verschwiegen hat: Befindet sich im bearbeiteten Basic-Programm ein Sprung auf eine nicht vorhandene Zeile, so wird dieser ohne Meldung ignoriert. Dies ist zwar nicht gerade ein Fehler, aber bei Unkenntnis eventuell doch prekär, denn die originale RENUMBER-Routine gibt in diesem Fall die Meldung UNRESOLVED REFERENCE ERROR aus.

Nach Einarbeitung in das Programm habe ich die vorgenannten Fehler beseitigt. Nur die letztgenannte Besonderheit der RENUMBER-Routine wurde aus Geschwindigkeitsgründen nicht geändert. Ferner wurde das Programm so gestaltet, daß es bei der Initialisierung nur noch einen Vektor aus der Vektoren-Tabelle verändert, nämlich $0304/$0305 (Umwandlung eines Basic-Befehls in Token). Tippen Sie Listing 7 im C 64-Modus mit dem MSE ab. Die Initialisierung der RENUMBER-Routine erfolgt mit SYS 4888. Die Syntax des neuen RENUMBER-Befehls ist dieselbe wie beim originalen und ist nur durch die Angabe des Endes des umzunumerierenden Bereichs erweitert.

Alle Befehlsparameter sind wie beim originalen RENUMBER-Befehl der Reihe nach optional, das heißt, es können von hinten her Parameter weggelassen werden. Beispiele:
RENUMBER (ohne Parameter): Neunumerierung von Anfang bis Ende mit der Schrittweite 10 und der Anfangszeilennummer 100.
RENUMBER 500: Neunumerierung von Anfang bis Ende mit der Schrittweite 10, die neue Anfangszeilennummer ist 500.
RENUMBER 111,5: Neunumerierung von Anfang bis Ende mit der Schrittweite 5, die neue Anfangszeilennummer ist 111.
RENUMBER 1000, 10, 300: Neunumerierung ab Zeile 300 bis Ende mit der Schrittweite 10, die neue Anfangszeilennummer ist 1000.
RENUMBER 40, 3, 100, 140: Neunumerierung von Zeile 100 bis 140 mit der Schrittweite 3 und einfügen ab Zeile 40. Bei diesem Beispiel wird der Zeilenbereich 100 bis 140 nach Zeile 40 v erschoben. Dabei darf diese Zeile nicht bereits existieren und der neue Bereich darf nicht in andere vorhandene Zeilen hineinreichen. Ist dies doch der Fall, wird die Fehlermeldung CROSSING LINE NUMBER ERROR ausgegeben und die Neunumerierung unterbleibt. Will man nach der Initialisierung der neuen RENUMBER-Routine die originale verwenden, so muß lediglich vor den Befehl ein »:« gesetzt werden.

(H. Büche/ef)

```
Name : renumber 4888       1300 1997
-----------------------------------
1300 : 4c 18 13 64 00 0a 00 00  fa
1308 : 00 00 00 01 0a 64 e8 10  b0
1310 : 00 00 00 03 27 40 ea ea  67
1318 : a2 23 a0 13 8e 04 03 8c  05
1320 : 05 03 60 20 0d 43 a2 ff  38
1328 : e8 bd 00 02 f0 f4 c9 20  4d
1330 : f0 f6 c9 f8 d0 ec a0 07  32
1338 : b9 03 13 99 00 11 88 10  36
1340 : f7 88 84 fc 84 fd a0 00  f7
1348 : 84 fa 84 fb 20 ec 18 b0  15
1350 : 28 8c 00 11 8d 01 11 20  46
1358 : ea 18 b0 18 8c 02 11 8d  b6
1360 : 03 11 20 ea 18 b0 0d 84  95
1368 : fa 85 fb 20 ea 18 b0 04  62
1370 : 84 fc 85 fd a0 ff c6 3e  35
1378 : 2c a0 00 20 df 18 78 20  d9
1380 : 0c 14 8d 01 ff 20 07 19  69
1388 : 20 5c 16 a5 2a 05 2b f0  6a
1390 : 6b 18 a9 04 65 2a 90 02  e0
1398 : e6 2b 0a 26 2b 0a 26 2b  4d
13a0 : 85 2a 38 ad 12 12 e5 2a  9c
13a8 : 85 28 ad 13 12 e5 2b 85  17
13b0 : 29 20 74 18 38 a5 28 e5  27
13b8 : 1a 85 22 a5 29 e5 1b 85  0b
13c0 : 23 90 4f 20 a3 17 a5 2b  e3
13c8 : 06 2a 2a 06 2a 0a aa e8  9e
13d0 : 86 26 e4 23 b0 3c a5 fa  80
13d8 : 05 fb d0 0c 20 60 18 b1  59
13e0 : 20 85 fa c8 b1 20 85 fb  c5
13e8 : 20 f8 16 f0 0c 10 25 20  3e
13f0 : c4 16 d0 20 20 a7 15 d0  2d
13f8 : 1b 20 52 14 20 09 19 8d  04
1400 : 03 ff 20 0c 14 8d 03 ff  46
1408 : 58 4c a2 4a 20 4f 4f 4c  cb
1410 : 82 4f a0 80 20 09 19 8d  3c
1418 : 03 ff 58 98 10 05 a2 10  18
1420 : 4c 3c 4d c0 04 d0 05 a2  16
1428 : 26 4c 3c 4d c0 06 f0 05  37
1430 : a2 0e 4c 3c 4d 20 81 92  75
1438 : 0d 3f 43 52 4f 53 53 49  6f
1440 : 4e 47 20 4c 49 4e 45 20  20
1448 : 4e 55 4d 42 45 52 00 4c  5c
1450 : a5 4d 20 74 18 20 6b 18  93
1458 : 85 18 85 1c 85 2f 18 8a  15
1460 : 85 19 65 26 85 1d 85 30  48
1468 : 20 1c 19 20 6b 18 a0 00  db
1470 : 84 be 84 bf b1 2f 91 16  73
1478 : c8 b1 2f 91 16 d0 07 c8  ac
1480 : 91 16 a8 91 16 60 c8 c8  92
1488 : b1 2f 88 aa b1 2f 20 69  30
1490 : 15 a0 02 91 16 c8 8a 91  9d
1498 : 16 c8 b1 2f 91 16 20 38  1f
14a0 : 18 b0 12 30 0b d0 f2 c8  30
14a8 : 20 94 18 20 a0 18 90 be  a7
14b0 : 20 0d 18 d0 e4 20 53 15  3e
14b8 : 90 e0 84 33 a2 00 95 18  f1
14c0 : e8 c8 b1 2f 20 5c 15 b0  f9
14c8 : f5 84 34 ca b5 18 85 24  e0
14d0 : a0 00 84 25 f0 1a b5 18  1d
14d8 : 85 27 c8 10 0f 18 a5 24  b6
14e0 : 79 0b 13 85 24 a5 25 79  4b
14e8 : 10 13 85 25 c6 27 10 ed  4a
14f0 : ca 10 e3 20 6d 15 b0 56  ae
14f8 : f0 54 85 24 86 25 a2 04  1c
1500 : a9 30 85 27 38 b0 06 84  32
1508 : 24 85 25 e6 27 a5 24 fd  41
1510 : 0b 13 a8 a5 25 fd 10 13  2c
1518 : b0 ed a5 27 95 18 ca d0  f4
1520 : df 18 a5 24 69 30 85 18  58
1528 : ac 34 00 20 94 18 a4 33  f5
1530 : 20 a0 18 a2 04 b5 18 c9  dd
1538 : 30 d0 03 ca d0 f7 a0 00  3a
1540 : b5 18 91 16 c8 ca 10 f8  3e
1548 : 20 a0 18 a0 00 2c a4 33  2d
1550 : 4c 9a 14 c8 b1 2f 91 16  0e
1558 : c9 20 f0 f7 38 e9 30 90  21
1560 : 07 c9 0a b0 02 38 60 18  78
1568 : 60 85 24 86 25 84 27 20  b8
1570 : 60 18 a0 00 f0 05 c8 d0  00
1578 : 02 e6 21 b1 20 aa c8 b1  4a
1580 : 20 c9 fa b0 1a c8 c8 e4  2e
1588 : 24 d0 eb c5 25 d0 e7 b1  a4
1590 : 20 aa 88 b1 20 a4 27 c5  ad
1598 : 24 d0 02 e4 25 18 60 a5  21
15a0 : 24 a6 25 a4 27 38 60 a5  f6
15a8 : fa 85 24 a5 fb 85 25 20  e3
15b0 : 98 16 85 2f 86 30 a6 fc  19
15b8 : a4 fd e8 d0 01 c8 86 24  68
15c0 : 84 25 20 98 16 85 31 86  51
15c8 : 32 38 e5 2f 85 35 8a e5  6e
15d0 : 30 85 36 38 a5 22 e5 35  c5
15d8 : a5 23 e5 36 b0 03 a0 04  fd
15e0 : 60 ad 00 11 85 24 ad 01  6b
15e8 : 11 85 25 20 98 16 85 33  c0
15f0 : 86 34 85 18 86 19 18 65  51
15f8 : 35 85 1c 8a 65 36 85 1d  a1
1600 : 20 74 18 18 65 35 85 37  e8
1608 : 8a 65 36 85 38 20 1c 19  aa
1610 : 38 a5 2f e5 33 a5 30 e5  90
1618 : 34 90 1a 18 a5 2f 65 35  f2
1620 : 85 2f a5 30 65 36 85 30  2b
1628 : 18 a5 31 65 35 85 31 a5  9b
1630 : 32 65 36 85 32 a2 05 b5  0b
1638 : 2f 95 18 ca 10 f9 20 1c  1b
1640 : 19 a5 31 85 18 a5 32 85  ab
1648 : 19 a5 37 85 1a a5 38 85  6d
1650 : 1b a5 2f 85 1c a5 30 85  75
1658 : 1d 20 1c 19 20 6b 18 a9  c1
1660 : 00 85 2a 85 2b a0 01 b1  7d
1668 : 16 f0 28 e6 2a d0 02 e6  dc
1670 : 2b a0 03 c8 b1 16 d0 fb  cc
1678 : c8 18 98 65 16 a6 17 90  33
1680 : 01 e8 a0 00 91 16 48 c8  9a
1688 : 8a 91 16 68 85 16 86 17  bf
1690 : 4c 65 16 a0 00 91 16 60  ce
1698 : 20 6b 18 d0 05 20 b5 16  e2
16a0 : f0 0d a0 02 b1 16 aa c8  87
16a8 : b1 16 20 b8 18 90 ee a5  91
16b0 : 16 a6 17 18 60 a0 00 b1  51
16b8 : 16 aa c8 b1 16 f0 04 86  92
16c0 : 16 85 17 60 20 60 18 ad  2b
16c8 : 00 11 85 24 ad 01 11 85  69
16d0 : 25 20 ac 18 70 1c f0 1d  19
16d8 : b0 05 20 85 18 90 f2 ad  f1
16e0 : 04 11 85 24 ad 05 11 85  a5
16e8 : 25 20 ac 18 70 04 90 05  bf
16f0 : f0 03 a0 00 60 a0 06 60  6e
16f8 : 20 60 18 a5 fa 85 24 a5  bb
1700 : fb 85 25 a9 00 aa 8e 06  d8
1708 : 11 8d 07 11 20 ac 18 70  6d
1710 : 4d f0 07 b0 4c 20 85 18  b9
1718 : 90 ec a5 fc 85 24 a5 fd  33
1720 : 85 25 ad 00 11 85 16 ad  94
1728 : 01 11 85 17 a0 03 a5 17  dd
1730 : 8d 05 11 91 20 88 a5 16  bf
1738 : 8d 04 11 91 20 18 6d 02  ba
1740 : 11 85 16 ad 03 11 65 17  cc
1748 : 85 17 b0 21 c9 fa b0 1d  1a
1750 : 20 85 18 20 ac 18 70 2b  e1
1758 : 90 d2 f0 d0 b0 12 a0 02  ca
1760 : 60 b1 20 85 fb 88 b1 20  5d
1768 : 85 fa 18 90 ad a0 04 60  33
1770 : ad 04 11 85 24 ad 05 11  fa
1778 : 85 25 20 ac 18 70 04 90  64
1780 : 1c f0 1a ad 07 11 ae 06  10
1788 : 11 d0 03 a8 f0 12 cd 01  b0
1790 : 11 90 0d d0 05 ec 00 11  21
1798 : 90 06 a0 cf 60 a0 d5 60  70
17a0 : a0 00 60 20 6b 18 20 60  15
17a8 : 18 84 2a 84 2b a0 00 84  de
17b0 : be 84 bf c8 b1 16 d0 0a  dc
17b8 : a9 ff a0 04 91 20 88 10  66
17c0 : fb 60 a0 03 b1 16 aa 88  fb
17c8 : b1 16 a0 00 91 20 c8 48  7a
17d0 : 8a 91 20 c8 68 91 20 c8  69
17d8 : 8a 91 20 20 85 18 a0 03  d9
17e0 : c8 b1 16 f0 22 20 38 18  59
17e8 : b0 07 10 f4 20 0d 18 d0  2b
17f0 : ef c8 b1 16 f0 11 c9 20  71
17f8 : f0 f7 20 5c 15 90 e2 e6  a7
1800 : 2a d0 dd e6 2b d0 d9 20  c7
1808 : b5 16 4c ad 17 a6 be d0  d4
1810 : 26 c9 89 90 22 f0 20 c9  4d
1818 : d7 b0 1c c9 d6 b0 18 c9  6e
1820 : d5 b0 14 c9 a7 b0 10 c9  5f
1828 : 91 d0 02 85 bf c9 8d b0  34
1830 : 06 c9 8c b0 02 c9 8a 60  ad
1838 : 48 a2 00 c9 22 d0 0b a5  2b
1840 : be 49 ff 85 be 86 bf 18  a3
1848 : 68 60 c9 3a d0 04 ea ea  49
1850 : 86 bf a6 bf f0 07 c9 2c  1e
1858 : d0 03 ea 68 60 18 68 60  9b
1860 : a0 00 a5 28 85 20 a5 29  b1
1868 : 85 21 60 a5 2d a6 2e 85  16
1870 : 16 86 17 60 18 ad 10 12  ef
1878 : ae 11 12 69 02 90 01 e8  db
1880 : 85 1a 86 1b 60 18 48 a9  53
1888 : 04 65 20 85 20 90 02 e6  54
1890 : 21 68 18 60 18 98 65 2f  32
1898 : 85 2f 90 03 e6 30 18 60  4a
18a0 : 18 98 65 16 85 16 90 03  72
18a8 : e6 17 18 60 a0 00 b1 20  3d
18b0 : aa c8 b1 20 c9 fb b0 0c  86
18b8 : c5 25 90 06 f0 02 b0 02  db
18c0 : e4 24 b8 60 2c 15 13 60  69
18c8 : c8 20 d6 18 c9 20 f0 f8  ac
18d0 : 60 a0 00 2c a0 01 8d 01  50
18d8 : ff b1 3d 8d 03 ff 60 18  93
18e0 : 98 65 3d 85 3d 90 02 e6  59
18e8 : 3e 60 a0 ff 20 c8 18 c9  bb
18f0 : 00 f0 13 c9 3a f0 0f c9  61
18f8 : 2c d0 01 c8 20 df 18 20  87
1900 : d7 77 20 15 88 18 60 18  39
1908 : 24 38 a2 20 b5 18 90 05  5d
1910 : bd 08 11 95 18 9d 08 11  79
1918 : ca 10 f1 60 38 a5 1a e5  58
1920 : 18 a8 a5 1b e5 19 48 38  12
1928 : a5 18 e5 1c aa a5 19 e5  de
1930 : 1d d0 05 8a d0 02 68 60  c7
1938 : 68 90 31 aa e8 84 1a 38  15
1940 : a9 00 e5 1a a8 84 1a 38  2d
1948 : a5 18 e5 1a 85 18 b0 02  96
1950 : c6 19 38 a5 1c e5 1a 85  ca
1958 : 1c b0 02 c6 1d b1 18 91  09
1960 : 1c c8 d0 f9 e6 19 e6 1d  61
1968 : ca d0 f2 60 84 18 aa e8  e8
1970 : 18 65 1d 85 1d 38 a5 1a  91
1978 : e5 18 85 1a b0 02 c6 1b  7a
1980 : 98 f0 07 b1 1a 91 1c 88  38
1988 : d0 f9 b1 1a 91 1c c6 1b  50
1990 : c6 1d ca d0 f2 60 20 fd  60
```

Listing 7.
Eine schnelle Renumber-Routine

14. POP RETURN

Wer kennt nicht das Problem, nach einem ungewollten GOSUB-Befehl wie zum Beispiel COLLISION nicht in das Hauptprogramm zurückkehren zu wollen. Hier hilft »POP.ASM« (Listing 8), das den letzten GOSUB-Befehl vergessen läßt. Wenn noch gar kein GOSUB-Befehl auftrat, gibt es die Fehlermeldung »RETURN WITHOUT GOSUB« aus.

```
Name : pop.asm                2000 2020
---------------------------------------
2000 : 68 68 a9 8d 20 aa 4f f0  2f
2008 : 05 a2 0c 4c 3c 4d 20 50  3a
2010 : 50 a0 05 20 59 50 4c 8f  5e
2018 : 52 ff 00 ff 00 ff 00 b1  cd
```

Listing 8. »POP.ASM« beseitigt die Rücksprungadresse eines vorangegangenen GOSUBs

Das Programm ist im Speicher mit dem MONITOR-Befehl »t« frei verschiebbar. Listing 9 demonstriert die Anwendung dieser Programmierhilfe. (K. H. Guckler/ef)

```
0 BLOAD "POP.ASM": BANK 15
1 PRINT "{DOWN}IM HAUPTPROGRAMM"
2 GOSUB 10
3 PRINT
4 GOTO 1
7 END
10 PRINT "{DOWN}NACH GOSUB 10 ..."
11 INPUT "(R) FUER RETURN ODER (=) FUER BEFEHL POP ";A
   $
12 IF A$<>"R" AND A$<>"=" THEN 11
13 IF A$="=" THEN SYS 8192
14 PRINT "{2SPACE}-> RETURN"
15 RETURN
```

Listing 9. »POP.Beispiel« demonstriert die Anwendung von POP.ASM

15. Großbuchstaben in der Grafik

»BIG LETTERS« (Listing 10) ist eine Basic-Erweiterung, die Zeichen in doppelter Höhe in bis zu zwei Farben auf dem Grafikbildschirm darstellt. Hierzu werden vom Programm fünf neue Befehle bereitgestellt. Im einzelnen handelt es sich hier um die Befehle SET, OUT, BIGON, BIGOFF und OFF. Letzterer dient dazu, die Erweiterung abzuschalten. Doch nun die Funktion der restlichen Befehle im einzelnen:

SET Zeile,Spalte,Farbe1,Farbe2

Der SET-Befehl legt Zeile, Spalte und Farbe der oberen und der unteren Zeichenhälfte für den OUT-Befehl fest. Alle Angaben können auch über Variablen erfolgen.

OUT "XXXXXXXXXXXXXXXXXXXXXXXXX"

Mit dem OUT-Befehl wird die nachfolgende Zeichenkette unter den mit SET angegebenen Daten auf dem Grafikbildschirm ausgegeben. Der Grafikbildschirm braucht dazu nicht eingeschaltet sein. Auch hier kann die Zeichenkette als Variable angegeben werden.

BIGON

Für diesen Befehl sind keine weiteren Angaben erforderlich. Er hat die Aufgabe, den oberen und unteren Bildschirmrand abzuschalten.

BIGOFF

Dieser Befehl bewirkt das Gegenteil von BIGON. Er schaltet den Bildschirmrand wieder ein.

Um Ihnen die Wirkungsweise der einzelnen Befehle zu verdeutlichen, liegt mit Listing 2 noch ein Demo-Listing vor, das Ihnen die Fähigkeiten der Erweiterung vor Augen führen soll.

Laden und starten

Nachdem Sie Listing 10 im C 64-Modus eingegeben haben, laden Sie es im C 128-Modus mit BLOAD "BIG LETTERS" von Diskette und starten es mit SYS 4864. Nun stehen Ihnen die oben beschriebenen Befehle zur Verfügung. Geben Sie das Demoprogramm (Listing 11) auf die gleiche Art und Weise ein. Das Abtippen mit dem Basic-Interpreter scheidet aus, da unser Druckprogramm (»Cursor-Translator«) die neuen Basic-Befehle nicht interpretieren kann.

(D. Bozza/ef)

```
Name : big letters           1300 1599
---------------------------------------
1300 : a9 58 a0 13 8d 0c 03 8c  be
1308 : 0d 03 a9 70 a0 13 8d 0e  04
1310 : 03 8c 0f 03 a9 8c a0 13  25
1318 : 8d 10 03 8c 11 03 a9 2b  26
1320 : a0 13 8d 00 03 8c 01 03  4c
1328 : 4c 4f 9f 8a 30 08 a9 1b  75
1330 : 8d 5a 15 20 83 15 4c 3f  c4
1338 : 4d 53 45 d4 4f 55 d4 42  92
1340 : 49 47 4f ce 42 49 47 4f  05
1348 : 46 c6 4f 46 c6 00 a7 13  bf
1350 : 18 14 2a 15 82 15 50 42  36
1358 : aa a9 13 a0 39 20 e2 43  56
1360 : 10 09 29 7f 18 69 27 a2  de
1368 : 00 90 02 38 8a 4c 21 43  4e
1370 : e0 ff f0 14 c9 27 90 10  47
1378 : c9 2c b0 0c e9 26 09 80  fa
1380 : aa a9 13 a0 39 4c 6a 51  1a
1388 : 38 4c cd 51 c9 02 90 14  9b
1390 : c9 2c b0 10 e9 26 0a aa  eb
1398 : bd 4f 13 48 bd 4e 13 48  f6
13a0 : 18 4c a9 4b 38 4c a9 4b  d5
13a8 : a9 00 8d 00 ff ad 21 d0  48
13b0 : 29 0f 85 fc 85 fd 20 f4  14
13b8 : 87 e0 18 b0 36 86 fa 20  8f
13c0 : ed 13 e0 29 b0 2d 86 fb  1b
13c8 : 20 ed 13 e0 10 b0 24 8a  ec
13d0 : 18 a5 fc 69 10 85 fc ca  de
13d8 : d0 f6 20 ed 13 e0 10 b0  c3
13e0 : 12 8a 18 a5 fd 69 10 85  69
13e8 : fd ca d0 f6 60 20 5c 79  c9
13f0 : 4c f4 87 a2 0e 6c 00 03  37
13f8 : e6 a8 d0 02 e6 a9 60 a0  25
1400 : 00 a2 04 b1 aa 91 a8 20  a3
1408 : f8 13 91 a8 e6 aa d0 02  0e
1410 : e6 ab 20 f8 13 ca d0 eb  96
1418 : 60 a5 eb 48 a5 ec 48 a9  85
1420 : 87 85 03 a9 7b 85 04 ad  af
1428 : 00 ff 8d de 02 a9 0f 85  1c
1430 : 02 20 cd 02 a5 06 85 fe  95
1438 : a9 00 a0 1c 85 ac 84 ad  b8
1440 : a6 fa f0 0e 18 a5 ac 69  95
1448 : 28 85 ac 90 02 e6 ad ca  14
1450 : d0 f2 18 a5 ac 65 fb 85  45
1458 : ac 90 02 e6 ad a9 00 a0  13
1460 : 20 85 a8 84 a9 a6 fb f0  9f
1468 : 0e 18 a5 a8 69 08 85 a8  3f
1470 : 90 02 e6 a9 ca d0 f2 a6  3c
1478 : fa f0 10 18 a5 a8 69 40  b7
1480 : 85 a8 90 02 e6 a9 e6 a9  68
1488 : ca d0 f0 ad 00 04 48 ad  49
1490 : 00 d8 48 a6 fe f0 17 a9  0a
1498 : 13 20 d2 ff a0 00 a9 24  69
14a0 : 8d aa 02 a2 7f 20 a2 02  df
14a8 : 20 d2 ff ae 00 04 68 8d  e4
14b0 : 00 d8 68 8d 00 04 a9 00  af
14b8 : a0 d0 85 aa 84 ab e0 00  a0
14c0 : f0 0e 18 a5 aa 69 08 85  93
14c8 : aa 90 02 e6 ab ca d0 f2  52
14d0 : a0 00 20 ff 13 18 a5 a8  52
14d8 : 69 38 85 a8 90 02 e6 a9  dc
14e0 : e6 a9 20 ff 13 a5 fc 91  18
14e8 : ac 18 a5 ac 69 28 85 ac  e6
14f0 : 90 02 e6 ad a5 fd 91 ac  da
14f8 : 38 a5 ac e9 28 85 ac b0  2e
1500 : 02 c6 ad e6 ac d0 02 e6  d5
1508 : ad 38 a5 a8 e9 40 85 a8  58
1510 : b0 02 c6 a9 c6 a9 e6 24  46
1518 : d0 02 e6 25 c6 fe f0 03  76
1520 : 4c 8b 14 68 a8 68 aa 18  ed
```

Listing 10. »BIG LETTERS«, eine kleine Basic-Erweiterung für große Buchstaben

```
1528 : 4c f0 ff a9 00 8d 00 ff  8e
1530 : 8d ff 3f 78 a9 5b 8d 14  70
1538 : 03 a9 15 cd 15 03 f0 18  6c
1540 : 8d 15 03 ad 11 d0 29 7f  0a
1548 : 8d 11 d0 8d 5a 15 a9 32  9d
1550 : 8d 12 d0 a9 81 8d 1a d0  de
1558 : 58 60 00 ad 19 d0 8d 19  17
1560 : d0 ad 12 d0 c9 f9 90 0d  6e
1568 : a9 13 8d 11 d0 a9 32 8d  5f
1570 : 12 d0 4c 33 ff ad 5a 15  65
1578 : 8d 11 d0 a9 f9 8d 12 d0  ed
1580 : 4c 65 fa ce 00 ff 78 a9  4c
1588 : 65 8d 14 03 a9 fa 8d 15  ec
1590 : 03 ad 5a 15 8d 11 d0 58  f8
1598 : 60 d0 02 e4 25 18 60 a5  5d
```

Listing 10. (Schluß)

```
Name : big letters demo  1c01 1edc
-----------------------------------
1c01 : 1c 1c 0a 00 8f 20 2a 2a  a5
1c09 : 2a 2a 2a 2a 2a 2a 2a 2a  09
1c11 : 2a 2a 2a 2a 2a 2a 2a 2a  11
1c19 : 2a 2a 00 37 1c 14 00 8f  c1
1c21 : 20 2a 20 42 49 47 20 4c  8f
1c29 : 45 54 54 45 52 20 20 44  85
1c31 : 45 4d 4f 20 2a 00 52 1c  19
1c39 : 1e 00 8f 20 2a 2a 2a 2a  30
1c41 : 2a 2a 2a 2a 2a 2a 2a 2a  41
1c49 : 2a 2a 2a 2a 2a 2a 2a 2a  49
1c51 : 00 6d 1c 28 00 8f 20 2a  65
1c59 : 20 45 49 4e 20 50 52 4f  a4
1c61 : 47 52 41 4d 4d 20 56 4f  99
1c69 : 4e 20 2a 00 88 1c 32 00  84
1c71 : 8f 20 2a 20 20 20 20 44  ab
1c79 : 49 52 4b 20 42 4f 5a 5a  7f
1c81 : 41 20 20 20 20 2a 00 a3  79
1c89 : 1c 3c 00 8f 20 2a 2a 2a  06
1c91 : 2a 2a 2a 2a 2a 2a 2a 2a  91
1c99 : 2a 2a 2a 2a 2a 2a 2a 2a  99
1ca1 : 2a 00 a9 1c 46 00 8f 00  5c
1ca9 : cd 1c 50 00 fe 11 22 42  1e
1cb1 : 49 47 20 4c 45 54 54 45  02
1cb9 : 52 53 22 2c 91 42 30 2c  07
1cc1 : 50 34 38 36 34 3a 9e 34  f8
1cc9 : 38 36 34 00 ee 1c 5a 00  63
1cd1 : e7 30 2c 31 3a e7 34 2c  0e
1cd9 : 36 3a de 31 2c 31 3a 58  f0
1ce1 : b2 31 35 3a 42 24 b2 22  15
1ce9 : 22 3a fe 29 00 10 1d 64  cb
1cf1 : 00 fe 27 32 2c 31 33 2c  f2
1cf9 : 32 2c 35 3a fe 28 22 54  38
1d01 : 48 45 20 42 49 47 20 4c  24
1d09 : 45 54 54 45 52 22 00 3a  e1
1d11 : 1d 6e 00 fe 27 36 2c 39  8d
1d19 : 2c 36 2c 31 34 3a fe 28  f3
1d21 : 22 28 43 29 20 49 4e 20  13
1d29 : 31 39 38 38 20 42 59 20  c6
1d31 : 44 42 2d 53 4f 46 54 22  09
1d39 : 00 67 1d 78 00 fe 27 38  48
1d41 : 2c 37 2c 38 2c 38 3a fe  86
1d49 : 28 22 2d 2d 2d 2d 2d 2d  bf
1d51 : 2d 2d 2d 2d 2d 2d 2d 2d  51
1d59 : 2d 2d 2d 2d 2d 2d 2d 2d  59
1d61 : 2d 2d 2d 2d 22 00 92 1d  bd
1d69 : 82 00 fe 27 31 33 2c 39  60
1d71 : 2c 35 2c 31 33 3a fe 28  ba
1d79 : 22 4e 41 4d 45 3a 20 2d  bd
1d81 : 2d 2d 2d 2d 2d 2d 2d 2d  81
1d89 : 2d 2d 2d 2d 2d 2d 2d 22  73
1d91 : 00 cc 1d 8c 00 fe 27 32  c9
1d99 : 32 2c 31 2c 38 2c 39 3a  f1
1da1 : fe 28 22 41 4c 4c 20 52  b0
1da9 : 49 47 48 54 53 20 52 45  3d
1db1 : 53 45 52 56 45 44 20 42  82
1db9 : 59 20 4d 41 52 4b 54 20  af
1dc1 : 26 20 54 45 43 48 4e 49  f7
1dc9 : 4b 22 00 e1 1d 96 00 fe  e6
1dd1 : 27 31 33 2c 58 2c 35 2c  f7
1dd9 : 31 33 3a 4e b2 31 30 00  72
1de1 : fb 1d a0 00 a1 41 24 3a  bc
1de9 : 8b 41 24 b1 22 22 a7 99  59
1df1 : 22 92 22 3b 3a 89 32 30  65
1df9 : 30 00 0b 1e aa 00 8b 4e  25
1e01 : b2 31 30 a7 99 22 12 22  84
1e09 : 3b 00 1f 1e b4 00 8b 4e  e6
1e11 : b2 32 30 a7 99 22 92 22  17
1e19 : 3b 3a 4e b2 30 00 33 1e  67
1e21 : be 00 8d 32 35 30 3a 4e  e3
1e29 : b2 4e aa 31 3a 89 31 36  f4
1e31 : 30 00 4f 1e c8 00 8b 41  36
1e39 : 24 b2 c7 28 31 33 29 a7  4e
1e41 : de 30 2c 31 3a 99 42 24  2a
1e49 : 3a fe 2a 3a 80 00 61 1e  9e
1e51 : d2 00 8b 41 24 b2 c7 28  76
1e59 : 32 30 29 a7 32 33 30 00  60
1e61 : 89 1e dc 00 8b 58 b3 33  e1
1e69 : 31 a7 fe 28 41 24 3a 42  d5
1e71 : 24 b2 42 24 aa 41 24 3a  bd
1e79 : 58 b2 58 aa 31 3a 89 31  03
1e81 : 35 30 3a d5 31 36 30 00  9d
1e89 : 9d 1e e6 00 8b 58 b3 31  9c
1e91 : 36 a7 31 36 30 3a d5 8d  f5
1e99 : 32 35 30 00 bd 1e f0 00  03
1ea1 : 58 b2 58 ab 31 3a 42 24  14
1ea9 : b2 c8 28 42 24 2c c3 28  15
1eb1 : 42 24 29 ab 31 29 3a 89  1d
1eb9 : 31 35 30 00 d4 1e fa 00  bb
1ec1 : 8b 58 b2 33 31 a7 fe 28  28
1ec9 : 22 20 22 3a d5 fe 28 22  05
1ed1 : 2d 22 00 da 1e 04 01 8e  8e
1ed9 : 00 00 00 01 fd 01 fd 01  db
```

Listing 11. »BIG LETTERS DEMO« zeigt die Fähigkeiten von BIG LETTERS

16. Der »Extended Color-Modus«

Haben Sie sich schon einmal gefragt, warum sich auf Ihrem C 128 nicht jedem Zeichen im Textmodus neben seiner Schrift- auch eine eigene Hintergrundfarbe zuweisen läßt? Hier ist eine einfache, aber äußerst wirkungsvolle Methode, die das Phänomen »Extended Color-Mode« (erweiterter Farbmodus) allen C 128-Fans eröffnet. Listing 12 zeigt, wie Sie bis zu vier verschiedene Hintergrundfarben auf den 40-Zeichen-Bildschirm zaubern.

Zeile 260: Durch Setzen von Bit 6 in Adresse 53265 (VIC-Register #27) aktiviert man den Extended Color-Modus (ECM). Das Ausschalten demonstriert Zeile 380.

Im ECM werden nur die ersten 64 verschiedenen Zeichen dargestellt. Dafür stehen vier Hintergrundfarben zur Auswahl (macht insgesamt wiederum 256 Zeichen-Codes).

Zeilen 270-300: Die vier Hintergrundfarben werden als Farbcodes (0 bis 15) in den Adressen 53281 bis 53284 untergebracht. Achtung: Die Farb-Codes sind VIC-spezifisch und liegen um 1 niedriger als die entsprechenden Basic 7.0-Farb-Codes beim COLOR-Befehl.

Zeilen 320 bis 360: Ausgabe der Texte. Die Hintergrundfarbe wird durch die Darstellungsart der Texte bestimmt:

ohne <SHIFT>, nicht revers : Farbe #0 (Adresse 53281)
mit <SHIFT>, nicht revers : Farbe #1 (Adresse 53282)
ohne <SHIFT>, revers : Farbe #2 (Adresse 53283)
mit <SHIFT>, revers : Farbe #3 (Adresse 53284)

Die Vordergrundfarbe ist wie gewohnt über COLOR 5,x oder über Steuerzeichen wählbar. Es ist noch darauf hinzuweisen, daß die mit <SHIFT> eingegebenen Zeichen so ausgegeben werden wie ohne <SHIFT>, dafür aber in einer anderen Farbe. Die Eingabe mit <SHIFT> dient nur zur Farbkennzeichnung. Der ECM läßt sich vielfältig einsetzen, natürlich vor allem dort, wo übersichtliche Bildschirmaufbauten gefragt sind. (F. Müller/ef)



```
100 REM ********************************
110 REM *                              *
120 REM *   BEISPIEL FUER DEN E.C.M.   *
130 REM *                              *
140 REM *     EXTENDED COLOR MODE      *
150 REM *                              *
160 REM ********************************
170 REM *                              *
180 REM * U.A. BASIEREND AUF TRICKS    *
190 REM *   AUS >>VOM C64 ZUM C128<<   *
200 REM *       (ISBN 3-89090-402-5)   *
210 REM *                              *
220 REM ********************************
230 :
240 :
```

Listing 12. Veranschaulicht den »Extended Color-Modus«

```
250 IF RWINDOW(2)=80 THEN PRINT "NUR 40-ZEICHEN-MODUS
    !": STOP
260 POKE 53265,PEEK(53265) OR 64: REM *** EXTENDED COL
    OR MODE WIRD AKTIVIERT
270 POKE 53281,5: REM HINTERGRUND #0 (NORMAL) = 5 (GRU
    EN)
280 POKE 53282,7: REM HINTERGRUND #1 (SHIFT)  = 7 (GEL
    B)
290 POKE 53283,15: REM HINTERGRUND #2 (REVERS) = 15 (H
    ELLGRAU)
300 POKE 53284,6: REM HINTERGRUND #3 (SHF/RVS)= 6 (BLA
    U)
310 SCNCLR : COLOR 5,1: CHAR ,0,0," DEMONSTRATION FUER
     EXTENDED COLOR MODE ",1
320 CHAR ,0,7,"SCHWARZ AUF GRUEN"
330 COLOR 5,3: CHAR ,0,8,"ROT{SHF.SPACE}AUF{SHF.SPACE}
    GELB{28SHF.SPACE}": REM *** SHIFT!
340 COLOR 5,7: CHAR ,0,9,"DUNKELBLAU AUF HELLGRAU{17SP
    ACE}",1
350 COLOR 5,2: CHAR ,0,10,"WEISS{SHF.SPACE}AUF{SHF.SPA
    CE}DUNKELBLAU{20SHF.SPACE}",1: REM SHIFT !
360 COLOR 5,9: CHAR ,0,20,"BY{SHF.SPACE}FLORIAN{SHF.SP
    ACE}MUELLER{17SHF.SPACE}TASTE"
370 PRINT CHR$(7): GET KEY A$: REM *** AUF TASTE WARTE
    N
380 POKE 53265,PEEK(53265) AND 191: REM *** EXTENDED C
    OLOR MODE AUSSCHALTEN
390 GRAPHIC 0,1: END
```

Listing 12. (Schluß)

17. Eigene Zeichensätze für den VDC

Den Zeichensatz im 40-Zeichen-Modus des C 128 zu verändern, stellt selbst den BASIC-Programmierer vor keine unlösbaren Probleme, zumal hierzu bereits genug Hilfsroutinen existieren. Anders hingegen sieht es aus, will man auch im 80-Zeichen-Modus in den Genuß eines eigenen Zeichensatzes kommen. Hierzu ist meines Wissens bisher keine brauchbare Routine veröffentlicht worden. Mit dem in Listing 14 abgedruckten Maschinenprogramm läßt sich diese Aufgabe ebenso schnell wie elegant lösen. Nach dem Aufruf mit SYS DEC("0B00") kopiert die Routine einen Zeichensatz (Listing 15 beinhaltet einen hübschen Demozeichensatz), der ab Adresse 8192 ($2000) in Bank 0 (im Grafikspeicher) liegen muß, in das VDC-RAM. Listing 16 demonstriert diesen Vorgang. Die neuen Zeichen stehen dann im Kleinschrift-Zeichensatz zur Verfügung ($3000 bis $3fff im VDC). Bei dem Programm handelt es sich um eine modifizierte ROM-Routine, die ab $FCE0C im Speicher steht. Achtung: Nach Drücken der ASCII/DIN-Taste wird der veränderte Zeichensatz durch das Betriebssystem mit dem Original überschrieben. Deshalb sollte man die ASCII/DIN-Taste durch POKE DEC ("0AC5"),128 vor dem Kopieren sperren.

Interessante Speicherstellen (ZS=Zeichensatz):

```
 $B10: HI-Byte der Zieladresse des ZS im VDC-RAM
Beispiel: POKE 2832,32   Beim nächsten Auf-
ruf von $B00 wird der ZS nach $2000 in den VDC
(Großschrift) kopiert.
 $B08: HI-Byte der Quelladresse im »normalen« RAM
Beispiel: Poke 2824,40   Durch den Aufruf von $B00
wird der Bereich ab 40*256=$2800 in den VDC über-
tragen.
ACHTUNG!: Inhalt von $B3C berücksichtigen
 $B3C: HI-Byte des ZS-Endes im »normalen« RAM
```

Im Normalfall sollte man diese Speicherstelle nach jeder Veränderung von $B08 durch POKE 2876,PEEK(2824)+8 re-initialisieren.

Im folgenden Listing 13 sehen Sie einen kommentierten Quelltext, den Sie im TOP-ASS-Format auch auf der Leser-Service-Diskette finden. Dank der zahlreichen Kommentare sind keine weiteren Erklärungen notwendig:

```
loc--code----number---mnemonic----------------------
             100  -;
             110  -; *** uebergibt zs ab zbase an vdc
             120  -;
             130  -.base $0b00
             140  -;
             150  -.define pntr        = $da
             160  -;
             170  -.define zbase       = $2000
             180  -.define zsatzend    = $2800
             190  -.define zsatz       = $3000
             200  -;
```

```
loc--code----number---mnemonic---------------------
0b00 a93e     210  -          lda #%00111110  ; alles ram,
0b02 8d00ff   220  -          sta $ff00  ; i/o ein
              230  -;
0b05 a900     240  -main      lda #<(zbase)
0b07 a220     250  -          ldx #>(zbase)
0b09 85da     260  -          sta pntr
0b0b 86db     270  -          stx pntr+1
              280  -;
0b0d a212     290  -          ldx #18   ; update adr hi
0b0f a930     300  -          lda #>(zsatz)
0b11 20420b   310  -          jsr writereg
0b14 e8       320  -          inx         ; und lo
0b15 a900     330  -          lda #<(zsatz)
0b17 20420b   340  -          jsr writereg
              350  -;
0b1a a000     360  -          ldy #0    ; zeichen aus zs
0b1c b1da     370  -loop1     lda (pntr),y   ;  holen
0b1e 20400b   380  -          jsr writedata  ; (a)-> vdc-ram
0b21 c8       390  -          iny
0b22 c008     400  -          cpy #8
0b24 90f6     410  -          bcc loop1
              420  -;
0b26 a900     430  -          lda #0          ; 8 nullbytes
0b28 20400b   440  -loop2     jsr writedata  ;  schreiben
0b2b 88       450  -          dey
0b2c d0fa     460  -          bne loop2
              470  -;
0b2e 18       480  -          clc
0b2f a5da     490  -          lda pntr       ; pointer in
0b31 6908     500  -          adc #8         ; zeichensatz
0b33 85da     510  -          sta pntr       ; auf naechtes
0b35 90e5     520  -          bcc loop1      ; zeichen
0b37 e6db     530  -          inc pntr+1
0b39 a5db     540  -          lda pntr+1     ; ende des
0b3b c928     550  -          cmp #>(zsatzend); zs er-
                              reicht ?
0b3d 90dd     560  -          bcc loop1    ; nein: weiterma-
                              chen
              570  -;
0b3f 60       580  -          rts
              590  -;
0b40 a21f     600  -writedata ldx #31       ; vdc data reg
0b42 8e00d6   610  -writereg  stx $d600
0b45 2c00d6   620  -writeloop bit $d600    ; auf vdc-status
0b48 10fb     630  -            bpl writeloop  ; warten
0b4a 8d01d6   640  -            sta $d601
0b4d 60       650  -            rts
              660  -;
```

Listing 13. Der Quelltext zu Z'SATZCOPY im TOP-ASS-Format

Es werden folgende Labels benutzt:

```
pntr         = $00da
main         = $0b05
loop1        = $0b1c
loop2        = $0b28
writedata    = $0b40
writereg     = $0b42
writeloop    = $0b45
zbase        = $2000
zsatzend     = $2800
zsatz        = $3000
```

Listing 13. (Schluß)

(C. Führner/ef)

```
Name : z'satzcopy.obj      0b00 0b4f
------------------------------------
0b00 : a9 3e 8d 00 ff a9 00 a2   be
0b08 : 20 85 da 86 db a2 12 a9   e1
0b10 : 30 20 42 0b e8 a9 00 20   5e
0b18 : 42 0b a0 00 b1 da 20 40   fb
0b20 : 0b c8 c0 08 90 f6 a9 00   28
0b28 : 20 40 0b 88 d0 fa 18 a5   cd
0b30 : da 69 08 85 da 90 e5 e6   09
0b38 : db a5 db c9 28 90 dd 60   55
0b40 : a2 1f 8e 00 d6 2c 00 d6   92
0b48 : 10 fb 8d 01 d6 60 30 8a   20
```

Listing 14. »Z'SATZ COPY.OBJ« kopiert einen Zeichensatz in den VDC

```
Name : chr-set              2000 2801
-------------------------------------
2000 : 1c 22 4a 56 4c 20 1e 00   c9
2008 : 00 00 1a 26 42 c6 39 00   93
2010 : 20 50 50 66 46 c5 38 00   ad
2018 : 00 00 0e 10 20 e0 1f 00   23
2020 : 02 02 1a 26 42 c6 39 00   ae
2028 : 00 00 1c 22 2c f0 1f 00   3a
2030 : 0c 12 14 18 30 d7 18 10   90
2038 : 00 00 1d 22 62 9f 22 1c   a8
2040 : 30 48 50 7c 42 c2 41 00   77
2048 : 08 00 08 18 28 c8 07 00   3a
2050 : 04 00 0c 34 47 9c 24 18   f8
2058 : 30 48 50 7c 42 ce 43 00   f8
2060 : 08 14 18 10 30 d0 0f 00   40
2068 : 00 00 54 6a aa aa 29 00   6f
2070 : 00 00 58 64 a4 a4 23 00   0f
2078 : 00 00 1c 22 43 a2 1c 00   7d
2080 : 00 00 5c 62 a2 a2 21 20   e8
2088 : 00 00 1a 26 46 ba 03 02   1e
2090 : 00 00 48 54 a3 a0 20 00   ed
2098 : 00 00 08 14 22 c2 1f 00   d2
20a0 : 08 08 3e 08 38 c8 07 00   23
20a8 : 00 00 42 42 42 c6 39 00   c0
20b0 : 00 00 46 46 42 a5 18 00   bc
20b8 : 00 00 43 4b 49 c9 36 00   ae
20c0 : 00 00 44 a8 90 28 47 00   4e
20c8 : 00 00 42 42 42 bf 22 1c   84
20d0 : 00 00 3e 42 8c 90 3f 00   f2
20d8 : 12 00 1a 26 42 c6 39 00   75
20e0 : 12 00 1c 22 43 a2 1c 00   f7
20e8 : 24 00 42 42 42 c6 39 00   24
20f0 : 18 24 44 4c 42 c2 4f 40   ad
20f8 : 00 00 10 20 7f 20 10 00   3a
2100 : 00 00 00 00 00 00 00 00   01
2108 : 08 08 08 08 00 00 08 00   38
2110 : 24 24 24 00 00 00 00 00   50
2118 : 24 24 7e 24 7e 24 24 00   0c
2120 : 1c 2a 28 1c 0a 2a 1c 00   41
2128 : 00 62 64 08 10 26 46 00   bf
2130 : 30 48 48 30 4a 44 3a 00   4c
2138 : 04 08 10 00 00 00 00 00   45
2140 : 04 08 10 10 10 08 04 00   a0
2148 : 20 10 08 08 08 10 20 00   f5
2150 : 08 2a 1c 3e 1c 2a 08 00   6f
2158 : 00 08 08 3e 08 08 00 00   e7
2160 : 00 00 00 00 00 08 08 10   e1
2168 : 00 00 00 7e 00 00 00 00   38
2170 : 00 00 00 00 00 18 18 00   92
2178 : 00 02 04 08 10 20 40 00   7f
2180 : 18 24 42 42 42 24 18 00   29
2188 : 04 0c 34 04 04 04 04 00   90
2190 : 3c 42 02 04 08 78 6e 00   ec
2198 : 3c 42 02 1c 02 42 3c 00   1d
21a0 : 0c 14 24 44 7e 04 04 00   60
21a8 : 7e 40 78 44 02 44 38 00   10
21b0 : 3c 42 40 7c 42 42 3c 00   d4
21b8 : 3e 42 04 1e 08 10 10 00   1e
21c0 : 3d 46 42 3c 42 42 3c 00   60
21c8 : 3c 42 46 3a 02 42 3c 00   21
21d0 : 00 00 08 00 00 08 00 00   13
21d8 : 00 00 08 00 00 08 08 10   5b
21e0 : 0e 18 30 60 30 18 0e 00   0e
21e8 : 00 00 7e 00 7e 00 00 00   70
21f0 : 70 18 0c 06 0c 18 70 00   74
21f8 : 3c 42 02 0c 10 00 10 00   99
2200 : 00 00 00 00 ff 00 00 00   00
2208 : 02 06 0a 3e 32 3f c2 00   80
2210 : 3c 42 22 2c 22 a2 5c 00   24
2218 : 06 0a 54 38 10 10 0f 00   fd
2220 : 3c 42 11 11 31 52 2c 00   3a
2228 : 1c 22 20 18 20 20 1f 00   e0
2230 : 7f 80 08 08 1c 48 30 00   b7
2238 : 38 48 f0 42 42 3f 22 1c   f8
2240 : 73 95 16 1c 34 54 23 00   f9
2248 : 36 0a 02 02 02 22 1c 00   e6
2250 : 36 0a 02 02 02 1f 22 1c   26
2258 : e2 22 34 38 24 a2 41 00   bc
2260 : 06 29 1e 08 28 58 37 00   a6
2268 : 09 09 1b 2d 29 49 c1 00   46
2270 : 09 1a 2a 2c 4c 48 88 00   c0
2278 : 19 2e 42 42 42 24 18 00   27
2280 : 3e 41 11 16 10 50 20 00   6a
2288 : 19 2e 42 42 4a 24 1b 00   c3
2290 : 3e 41 11 1e 14 52 21 00   cf
2298 : 06 29 1e 08 08 48 30 00   3f
22a0 : 7f 80 08 08 08 48 30 00   e6
22a8 : 42 22 22 22 22 22 1d 00   70
22b0 : 46 27 22 22 22 14 08 00   3a
22b8 : 86 47 42 52 52 5a 64 00   46
22c0 : 61 92 14 08 18 24 47 00   30
22c8 : 42 22 22 22 23 1e 22 1c   cc
22d0 : 3f 41 02 1f 08 78 6f 00   16
22d8 : 0b 06 0a 3e 32 3f c2 00   59
22e0 : 5a 24 42 42 42 24 18 00   cb
22e8 : 24 42 22 22 22 22 1d 00   a2
22f0 : 00 00 01 3e 54 14 22 00   67
22f8 : cc 66 33 99 cc 66 33 99   f7
2300 : 00 00 00 00 00 00 00 00   01
2308 : f0 f0 f0 f0 f0 f0 f0 f0   07
2310 : 00 00 00 00 ff ff ff ff   10
2318 : ff 00 00 00 00 00 00 00   18
2320 : 00 00 00 00 00 00 00 ff   20
2328 : 80 80 80 80 80 80 80 80   28
2330 : aa 55 aa 55 aa 55 aa 55   85
2338 : 01 01 01 01 01 01 01 01   38
2340 : 00 00 00 00 aa 55 aa 55   eb
2348 : 99 33 66 cc 99 33 66 cc   14
2350 : 03 03 03 03 03 03 03 03   50
2358 : 08 08 08 08 0f 08 08 08   c9
2360 : 00 00 00 00 0f 0f 0f 0f   24
2368 : 08 08 08 08 0f 00 00 00   68
2370 : 00 00 00 00 f8 08 08 08   70
2378 : 00 00 00 00 00 00 ff ff   78
2380 : 00 00 00 00 0f 08 08 08   e2
2388 : 08 08 08 08 ff 00 00 00   97
2390 : 00 00 00 00 ff 08 08 08   01
2398 : 08 08 08 08 f8 08 08 08   a7
23a0 : c0 c0 c0 c0 c0 c0 c0 c0   9f
23a8 : e0 e0 e0 e0 e0 e0 e0 e0   a7
23b0 : 07 07 07 07 07 07 07 07   b0
23b8 : ff ff 00 00 00 00 00 00   b7
23c0 : ff ff ff 00 00 00 00 00   bf
23c8 : 00 00 00 00 00 ff ff ff   c8
23d0 : 01 02 44 48 50 60 40 00   f6
23d8 : 00 00 00 00 f0 f0 f0 f0   15
23e0 : 0f 0f 0f 0f 00 00 00 00   1d
23e8 : 08 08 08 08 f8 00 00 00   87
23f0 : f0 f0 f0 f0 00 00 00 00   b3
23f8 : f0 f0 f0 f0 0f 0f 0f 0f   7e
2400 : e3 dd b5 a9 b3 df e1 ff   36
2408 : ff ff e5 d9 bd 39 c6 ff   7c
2410 : df af af 99 b9 3a c7 ff   72
2418 : ff ff f1 ef df 1f e0 ff   0c
2420 : fd fd e5 d9 bd 39 c6 ff   91
2428 : ff ff e3 dd d3 0f e0 ff   15
2430 : f3 ed eb e7 cf 28 e7 ef   cf
2438 : ff ff e2 dd 9d 60 dd e3   c7
2440 : cf b7 af 83 bd 3d be ff   08
2448 : f7 ff f7 e7 d7 37 f8 ff   55
2450 : fb ff f3 cb b8 63 db e7   a7
2458 : cf b7 af 83 bd 31 bc ff   b7
2460 : f7 eb e7 ef cf 2f f0 ff   7f
2468 : ff ff ab 95 55 55 d6 ff   60
2470 : ff ff a7 9b 5b 5b dc ff   d0
2478 : ff ff e3 dd bc 5d e3 ff   72
2480 : ff ff a3 9d 5d 5d de df   17
2488 : ff ff e5 d9 b9 45 fc fd   f1
2490 : ff ff b7 ab 5c 5f df ff   32
2498 : ff ff f7 eb dd 3d e0 ff   5d
24a0 : f7 f7 c1 f7 c7 37 f8 ff   1c
24a8 : ff ff bd bd bd 39 c6 ff   8f
24b0 : ff ff b9 b9 bd 5a e7 ff   a3
24b8 : ff ff bc b4 b6 36 c9 ff   c1
24c0 : ff ff bb 57 6f d7 b8 ff   31
24c8 : ff ff bd bd bd 40 dd e3   0b
24d0 : ff ff c1 bd 73 6f c0 ff   ad
24d8 : ed ff e5 d9 bd 39 c6 ff   3a
24e0 : ed ff e3 dd bc 5d e3 ff   c8
24e8 : db ff bd bd bd 39 c6 ff   ab
24f0 : e7 db bb b3 bd 3d b0 bf   32
24f8 : ff ff ef df 80 df ef ff   b5
2500 : ff ff ff ff ff ff ff ff   ff
2508 : f7 f7 f7 f7 ff ff f7 ff   d7
2510 : db db db ff ff ff ff ff   d0
2518 : db db 81 db 81 db db ff   23
2520 : e3 d5 d7 e3 f5 d5 e3 ff   fe
2528 : ff 9d 9b f7 ef d9 b9 ff   90
2530 : cf b7 b7 cf b5 bb c5 ff   13
2538 : fb f7 ef ff ff ff ff ff   2b
2540 : fb f7 ef ef ef f7 fb ff   df
```

Listing 15. »CHR-SET« ist ein Demo-Zeichensatz für Z'SATZ COPY.OBJ, der handgeschriebener Schrift nahekommt

```
2548 : df ef f7 f7 f7 ef df ff  9a
2550 : f7 d5 e3 c1 e3 d5 f7 ff  30
2558 : ff f7 f7 c1 f7 f7 ff ff  c8
2560 : ff ff ff ff ff f7 f7 ef  de
2568 : ff ff ff 81 ff ff ff ff  97
2570 : ff ff ff ff ff e7 e7 ff  4d
2578 : ff fd fb f7 ef df bf ff  71
2580 : e7 db bd bd bd db e7 ff  d6
2588 : fb f3 cb fb fb fb fb ff  7f
2590 : c3 bd fd fb f7 87 91 ff  33
2598 : c3 bd fd e3 fd bd c3 ff  12
25a0 : f3 eb db bb 81 fb fb ff  df
25a8 : 81 bf 87 bb fd bb c7 ff  3f
25b0 : c3 bd bf 83 bd bd c3 ff  8b
25b8 : c1 bd fb e1 f7 ef ef ff  52
25c0 : c2 b9 bd c3 bd bd c3 ff  1f
25c8 : c3 bd b9 c5 fd bd c3 ff  6e
25d0 : ff ff f7 ff ff f7 ff ff  8c
25d8 : ff ff f7 ff ff f7 f7 ef  54
25e0 : f1 e7 cf 9f cf e7 f1 ff  b1
25e8 : ff ff 81 ff 81 ff ff ff  5f
25f0 : 8f e7 f3 f9 f3 e7 8f ff  6c
25f8 : c3 bd fd f3 ef ff ef ff  56
2600 : ff ff ff ff 00 ff ff ff  ff
2608 : fd f9 f5 c1 cd c0 3d ff  8f
2610 : c3 bd dd d3 dd 5d a3 ff  fb
2618 : f9 f5 ab c7 ef ef f0 ff  32
2620 : c3 bd ee ee ce ad d3 ff  05
2628 : e3 dd df e7 df df e0 ff  6f
2630 : 80 7f f7 f7 e3 b7 cf ff  a8
2638 : c7 b7 0f bd bd c0 dd e3  77
2640 : 8c 6a e9 e3 cb ab dc ff  86
2648 : c9 f5 fd fd fd dd e3 ff  a9
2650 : c9 f5 fd fd fd e0 dd e3  79
2658 : 1d dd cb c7 db 5d be ff  f3
2660 : f9 d6 e1 f7 d7 a7 c8 ff  19
2668 : f6 f6 e4 d2 d6 b6 3e ff  89
2670 : f6 e5 d5 d3 b3 b7 77 ff  1f
2678 : e6 d1 bd bd bd db e7 ff  c8
2680 : c1 be ee e9 ef af df ff  95
2688 : e6 d1 bd bd b5 db e4 ff  4c
2690 : c1 be ee e1 eb ad de ff  50
2698 : f9 d6 e1 f7 f7 b7 cf ff  f0
26a0 : 80 7f f7 f7 f7 b7 cf ff  59
26a8 : bd dd dd dd dd dd e2 ff  df
26b0 : b9 d8 dd dd dd eb f7 ff  25
26b8 : 79 b8 bd ad ad a5 9b ff  29
26c0 : 9e 6d eb f7 e7 db b8 ff  4f
26c8 : bd dd dd dd dc e1 dd e3  c3
26d0 : c0 be fd e0 f7 87 90 ff  89
26d8 : f4 f9 f5 c1 cd c0 3d ff  56
26e0 : a5 db bd bd bd db e7 ff  f4
26e8 : db bd dd dd dd dd e2 ff  2d
26f0 : ff ff fe c1 ab eb dd ff  78
26f8 : 33 99 cc 66 33 99 cc 66  f8
2700 : ff ff ff ff ff ff ff ff  ff
2708 : 0f 0f 0f 0f 0f 0f 0f 0f  08
2710 : ff ff ff ff 00 00 00 00  0f
2718 : 00 ff ff ff ff ff ff ff  18
2720 : ff ff ff ff ff ff ff 00  1f
2728 : 7f 7f 7f 7f 7f 7f 7f 7f  28
2730 : 55 aa 55 aa 55 aa 55 aa  da
2738 : fe fe fe fe fe fe fe fe  37
2740 : ff ff ff ff 55 aa 55 aa  94
2748 : 66 cc 99 33 66 cc 99 33  7b
2750 : fc fc fc fc fc fc fc fc  4f
2758 : f7 f7 f7 f7 f0 f7 f7 f7  e6
2760 : ff ff ff ff f0 f0 f0 f0  9b
2768 : f7 f7 f7 f7 f0 ff ff ff  67
2770 : ff ff ff ff 07 f7 f7 f7  6f
2778 : ff ff ff ff ff ff 00 00  77
2780 : ff ff ff ff f0 f7 f7 f7  1d
2788 : f7 f7 f7 f7 00 ff ff ff  78
2790 : ff ff ff ff 00 f7 f7 f7  1e
2798 : f7 f7 f7 f7 07 f7 f7 f7  88
27a0 : 3f 3f 3f 3f 3f 3f 3f 3f  a0
27a8 : 1f 1f 1f 1f 1f 1f 1f 1f  a8
27b0 : f8 f8 f8 f8 f8 f8 f8 f8  af
27b8 : 00 00 ff ff ff ff ff ff  b8
27c0 : 00 00 00 ff ff ff ff ff  c0
27c8 : ff ff ff ff ff 00 00 00  c7
27d0 : fe fd bb b7 af 9f bf ff  aa
27d8 : ff ff ff ff 0f 0f 0f 0f  9a
27e0 : f0 f0 f0 f0 ff ff ff ff  a2
27e8 : f7 f7 f7 f7 07 ff ff ff  48
27f0 : 0f 0f 0f 0f ff ff ff ff  2d
27f8 : 0f 0f 0f 0f f0 f0 f0 f0  71
2800 : 44 03 fd 0b 7d 03 fd 03  95
```

Listing 15. »CHR.SET.« (Schluß)

```
10 REM  BEISPIELPROGRAMM ZUR BENUTZUNG DER ROUTINE "Z'
   SATZCOPY.OBJ"
20 REM   LAEUFT AM 80-ZEICHEN BILDSCHIRM
30 :
40 BLOAD "Z'SATZCOPY.OBJ" : REM  MASCHINENROUTINE NACH
   $B00 NACHLADEN
50 :
60 GRAPHIC 1                : REM  PLATZ FUER ZEICHENSAT
   Z IN BANK 0 RESERVIEREN
70 BLOAD "CHR-SET" ON B0,P 8192 : REM  ZS NACH $2000 N
   ACHLADEN
80 :
90 POKE DEC("AC5"),128      : REM  ASCII/DIN - TASTE SPE
   RREN
100 :
110 SYS DEC("B00")          : REM  RUFT ROUTINE AUF
120 :
130 GRAPHIC CLR             : REM  GRAPHIK-SPEICHER WIED
    ER FREIGEBEN
140 :                       : REM  (NICHT UNBEDINGT NOT
    WENDIG)
150 PRINT "{CLR}LIST": LIST
```

Listing 16. Ein Demo für Z'SATZ COPY.OBJ

18. Kleinbuchstaben bei CHAR

Sollten Sie schon einmal versucht haben, mit dem CHAR-Befehl des Basic 7.0 den Groß-/Kleinschrift-Zeichensatz zu benutzen, so werden Sie festgestellt haben, daß der Computer immer auf den Groß-/Grafik-Zeichensatz zugreift. Drückt man an entsprechender Stelle im auszugebenden Text <CONTROL N> (ein kleines, reverses »n« erscheint), so wird der folgende Text im Groß-/Kleinschrift-Modus ausgegeben.

Beispiel (Unterstrichen = invers): CHAR1,0,0,"TESTnTEST"

Ergebnis: TESTtest

Durch die Eingabe eines großen, reversen »N« schalten Sie wieder auf den Groß-/Grafik-Zeichensatz um. Geben Sie dazu zunächst »KEY 1,CHR$(142)« ein (oder jede andere Funktionstaste). Durch Drücken von F1 im auszugebenden Text schalten Sie in den ursprünglichen Zeichensatz zurück. Beachten Sie auch, daß bei jedem neuen CHAR-Befehl (im Grafik-Modus) zurückgeschaltet wird.

(J. H. Schulz/ef)

19. Paint Magic und der C 128

Mit Paint Magic aus dem 64'er Sonderheft 23 (Grafik) erstellen Sie vielleicht schon anspruchsvolle Grafiken. Wir wollen nun diese Bilder auch im C 128-Modus verwenden.

Zur Erinnerung: Paint Magic speichert die Bilder als Programmdateien zusammen mit einer Assembler-Routine zur Darstellung der Bilder. Diese Routine interessiert uns aber nicht. Wichtig sind nur die Speicherbereiche, in denen die Bilddaten liegen. Bei Paint Magic-Bildern liegt die Bitmap von $4000 bis $5F40 und das Farb-RAM von $6000 bis $63E8. Die Werte der Farben für Hintergrund, Multicolor und den Rahmen stehen von $5F40 bis $5F44. Beim C 128 liegt die Bitmap ab $2000 und das Farb-RAM beginnt bei $1C00.

Um Paint Magic-Bilder direkt in den Computer laden und ansehen zu können, muß man allerdings eine Kleinigkeit beachten: Nach dem Einschalten des Grafikmodus lassen sich die Multicolorfarben nicht mehr ändern. Wir können also nicht einfach den Grafikbildschirm löschen, ein Bild laden und danach die Farben setzen. Aber man kann sich mit einem kleinen Trick behelfen. Wir laden das Bild in Bank 1,

setzen dann die Farben und kopieren es zum Schluß in Bank 0. Da das Kopieren von einer Bank in eine andere in Basic nur sehr langsam vonstatten geht, ist es von Vorteil, den eingebauten Monitor zu benutzen. Die Grundlagen zur Benutzung der Monitor-Routinen von Basic aus lesen Sie am besten in der Ausgabe 4/87 des 64'er-Magazins nach.

Nun möchte man vielleicht die Bilder in eigene Programme einbauen. Dazu ist es sinnvoll, die Bilder gleich im C128-Format auf Diskette zu speichern. Dadurch können die Bilder dann direkt in den Grafikspeicher geladen werden.

Noch einmal kurz die einzelnen Schritte zur Konvertierung von Paint Magic-Bildern:

1. Grafikbildschirm einschalten und löschen:
   `GRAPHIC 3,1`
2. Paint Magic-Bild in Bank 1 laden:
   `BLOAD"Name",ON B1`
3. Im Monitor die Werte der Adressen $15F40 bis $15F44 notieren.
4. Mit dem Monitor die Bitmap verschieben:
   `T 14000 15F40 2000`
5. Mit dem Monitor das Farb-RAM verschieben:
   `T 16000 163E8 1C00`
6. Und schließlich das Bild speichern:
   `S"Neuer Name",08,1C00,3F40`

Vor dem Laden dieser Bilder müssen nur noch die Farben gesetzt und der Grafikmodus eingeschaltet werden. Für die Farben sind die notierten Werte einzusetzen. Und zwar:

COLOR 0 , Wert aus $15F40
COLOR 1 , Wert aus $15F41
COLOR 2 , Wert aus $15F42
COLOR 3 , Wert aus $15F43
COLOR 4 , Wert aus $15F44

Das Programm »PAINTMAGIC-128« (Listing 17) erledigt diese Aufgabe für Sie. Sie werden nach dem Namen des Bildes gefragt, das geladen werden soll. Wenn Sie »$« eingeben, erscheint das Directory der Diskette. Danach zeigt Ihnen der Computer die Farbwerte und fragt, ob Sie das Bild im C128-Format speichern wollen, oder ob Sie vielleicht nur die Bitmap speichern wollen. Dies ist bei mehreren Bildern mit identischem Farb-RAM sinnvoll, zum Beispiel bei Computer-Filmen oder -Animationen.

Sicher ist Ihnen aufgefallen, daß die Bilder nur noch 36 Blocks auf Diskette belegen. Der 37. Block des Originalbildes wird von der Routine zur Darstellung der Bilder eingenommen. Sie fällt bei dem neuen Format weg, und die Bilder benötigen etwas weniger Speicherplatz.(T. Gudella/ef)

```
1 GRAPHIC 1,1: GRAPHIC 5,1
2 INPUT "{CTRL+N,2DOWN}BILDNAME ";B$
3 IF B$="$" THEN DIRECTORY : GOTO 2
4 IF B$="" THEN 2
5 BLOAD (B$),ON B1
6 PRINT : BANK 1: FOR J=0 TO 4: COLOR J,PEEK(24384+J)+
  1: PRINT "COLOR";J;",";RCLR(J): NEXT : GRAPHIC 3,1:
  BANK 0
7 FOR I=1 TO 2
8 N$="T 14000 15F40 2000"
9 IF I=1 THEN N$="T 16000 163E8 1C00"
10 FOR J=1 TO 18: POKE 511+J,ASC(MID$(N$,J,1)): NEXT
11 POKE 511+J,0: ZL=PEEK(804): ZH=PEEK(805): POKE 804,
   108: POKE 805,151
12 BANK 15: SYS DEC("B0A6"): POKE 804,ZL: POKE 805,ZH:
    NEXT
13 GRAPHIC 5
14 INPUT "{RVSON}N{RVOFF}ICHT ABSPEICHERN, {RVSON}M{RV
   OFF}IT FARBRAM ABSPEICHERN, NUR {RVSON}B{RVOFF}ITMA
   P SPEICHERN";G$
15 IF G$="N" THEN END
16 IF G$<>"M" AND G$<>"B" THEN 14
17 INPUT "{3DOWN}BITTE DISK EINLEGEN, BILDNAME";B$
18 IF G$="M" THEN BSAVE(B$),P7168 TO P16193
19 IF G$="B" THEN BSAVE(B$),P8192 TO P16193
```

Listing 17. »Paint Magic-128« konvertiert Paint Magic-Bilder ins C 128-Format

20. Super-Hardcopy

Super-Hardcopy (Listing 18) funktioniert auf allen 8-Nadel-Druckern wie Star oder Epson. Das Programm erlaubt, Hardcopies vom Text- und Grafikbildschirm zu nehmen und diese beliebig nach x- und y-Achse zu vergrößern. Darüber hinaus können auch Texthardcopies mit eigenen Zeichensätzen gemacht werden.

Syntax: SYS 4867, Modus, Vergrößerung, Druckmodus
Modus 1 := Hardcopy vom Textbildschirm,
Modus <>1:= alle anderen Eingaben ergeben eine Grafik-Hardcopy!)
Vergrößerung:
Die oberen 4 Bits geben die x-Vergrößerung,
die unteren 4 Bits geben die y-Vergrößerung an.
Dieser Wert kann folgendermaßen berechnet werden: X * 16 + Y. (Ist die Angabe »0«, wird die Hardcopy in Originalgröße gedruckt).
Druckmodus:
Dieser Wert gibt den Druckmodus an und kann aus dem Druckerhandbuch entnommen werden.

Bei Epson LX 800 und Star NL-10 lauten die Druckmodi wie folgt:
0 oder 75: Normal density (480 dots);
76: Double density (960 dots);
89: double density with double speed;
90: Quadruple density (1920 dots);
Einige Beispiele:
SYS 4867,10,0 (Hardcopy des Textbildschirms in Originalgröße und Normal density)
SYS 4867,0,3*16+3,76 oder SYS 4867,0,51,76
Hardcopy des Grafik-Bildschirms,
Vergrößerung: dreifach x und y,
Bemerkung: 3 * 16 + 3 = 51;
x * 16 + y = Vergrößerung.
Texthardcopy mit eigenen Zeichensätzen:
SYNTAX: SYS 5442, Startlo, Starthi

Die Parameter »Startlo« und »Starthi« geben die Startadresse des eigenen Zeichensatzes an. Nach Eingabe dieses Befehls greift das Hardcopy-Programm auf diesen Zeichensatz zu.

Beispiele: SYS 5442,0,32 schaltet auf einen Zeichensatz ab $2000 (= 8192) um.
SYS 5442,0,216 schaltet auf den DIN-Zeichensatz um,
SYS 5442,0,208 schaltet auf den ASCII-Zeichensatz.

Listing 19 paßt Super-Hardcopy an den Star NL-10 oder den Epson LX-800 an, Listing 20 druckt eine Demo-Hardcopy. (E. Aitenbichler/ef)

```
Name : super-hardcopy    1303 1551
----------------------------------
1303 : 8d 4c 15 8e 53 15 8c 56  8a
130b : 15 ad 56 15 c9 00 d0 05  19
```

Listing 18. »SUPER-HARDCOPY« Flexible Hardcopies im 40-Zeichen-Modus

```
1313 : a9 4b 8d 56 15 ad 53 15  c6
131b : c9 00 d0 05 a9 11 8d 53  b9
1323 : 15 20 25 15 a9 00 85 fa  db
132b : 8d 54 15 8d 55 15 a9 20  be
1333 : 85 fb a9 00 85 fc a9 04  0f
133b : 85 fd a9 00 8d 02 13 20  9f
1343 : ce 13 20 43 14 a9 00 8d  b5
134b : 52 15 20 2b 14 a9 00 8d  3f
1353 : 00 13 20 5d 14 a2 00 a9  3a
135b : 00 9d 58 15 e8 e0 08 d0  3a
1363 : f6 a2 00 a0 00 b9 60 15  38
136b : 8d 01 13 bd c6 13 2c 01  ad
1373 : 13 f0 0a 18 bd 58 15 79  6a
137b : c6 13 9d 58 15 c8 c0 08  e8
1383 : d0 e3 e8 e0 08 d0 dc 20  56
138b : f4 13 ee 00 13 20 13 14  6b
1393 : ad 00 13 c9 28 d0 bb 20  77
139b : e1 ff f0 24 a9 0d 20 d2  66
13a3 : ff ee 52 15 ad 52 15 cd  ae
13ab : 57 15 d0 0e ee 02 13 ad  29
13b3 : 02 13 c9 19 d0 8f 20 0a  f2
13bb : 14 60 20 b9 14 4c 4d 13  3e
13c3 : 4c 0a 14 80 40 20 10 08  7f
13cb : 04 02 01 a9 01 a2 04 a0  bc
13d3 : 00 20 ba ff a9 00 20 bd  28
13db : ff 20 c0 ff a2 01 20 c9  61
13e3 : ff a9 1b 20 d2 ff a9 41  d8
13eb : 20 d2 ff a9 08 20 d2 ff  76
13f3 : 60 a2 00 a0 00 bd 58 15  32
13fb : 20 d2 ff c8 cc 53 15 d0  fb
1403 : f4 e8 e0 08 d0 ed 60 20  e3
140b : cc ff a9 01 20 c3 ff 60  42
1413 : 18 a5 fa 69 08 85 fa a5  cd
141b : fb 69 00 85 fb e6 fc a5  b2
1423 : fc c9 00 d0 02 e6 fd 60  2e
142b : a9 1b 20 d2 ff ad 56 15  b5
1433 : 20 d2 ff ad 54 15 20 d2  86
143b : ff ad 55 15 20 d2 ff 60  62
1443 : a2 00 18 ad 54 15 69 40  b5
144b : 8d 54 15 ad 55 15 69 01  a3
1453 : 8d 55 15 e8 ec 53 15 d0  4d
145b : e9 60 a9 00 8d 4f 15 a9  da
1463 : 00 8d 4e 15 a2 00 ac 4e  da
146b : 15 20 d4 14 ac 4f 15 99  15
1473 : 68 15 ee 4f 15 e8 ec 57  06
147b : 15 d0 eb ee 4e 15 ad 4e  b2
1483 : 15 c9 08 d0 df a9 68 85  91
148b : fe a9 15 85 ff ad 52 15  35
1493 : c9 00 f0 15 a2 00 18 a5  11
149b : fe 69 08 85 fe a5 ff 69  f0
14a3 : 00 85 ff e8 ec 52 15 d0  da
14ab : ed a0 00 b1 fe 99 60 15  87
14b3 : c8 c0 08 d0 f6 60 38 a5  96
14bb : fa e9 40 85 fa a5 fb e9  0b
14c3 : 01 85 fb 38 a5 fc e9 28  c7
14cb : 85 fc a5 fd e9 00 85 fd  a8
14d3 : 60 ad 4c 15 c9 01 f0 03  2e
14db : b1 fa 60 8e 4a 15 8c 4b  09
14e3 : 15 a0 00 b1 fc 8d 4d 15  1a
14eb : a2 00 86 fe a2 d0 86 ff  d9
14f3 : c9 00 f0 15 a2 00 18 a5  71
14fb : fe 69 08 85 fe a5 ff 69  50
1503 : 00 85 ff e8 ec 4d 15 d0  12
150b : ed 18 a5 fe 6d 4b 15 85  de
1513 : fe a5 ff 69 00 85 ff a2  82
151b : 0e a9 fe 20 74 ff ae 4a  58
1523 : 15 60 ad 53 15 a8 a2 00  5f
152b : 38 ad 53 15 e9 10 8d 53  ad
1533 : 15 e8 b0 f4 ca 8e 53 15  20
153b : 98 29 0f 8d 57 15 60 8d  98
1543 : ec 14 8e f0 14 60 00 00  3f
154b : 07 01 20 08 08 1d 12 d0  2f
```

Listing 18. »SUPER-HARDCOPY« (Schluß)

```
10 BLOAD "SUPER-HARDCOPY"
20 SCNCLR : PRINT "STAR NL-10{3SPACE}: 8"
30 PRINT "EPSON LX-800 : 7"
40 PRINT "{DOWN}ZEILENABSTAND BEI GRAFIKDRUCK :";PEEK(
5103)
50 CHAR 1,30,3: INPUT Z: POKE 5103,Z
60 A$="SUPER-HARDCOPY": SCRATCH (A$)
70 BSAVE (A$),P4867 TO P5456
```

Listing 19. Anpassung der Super-Hardcopy an den NL-10 oder LX 800

```
5 BLOAD "SUPER-HARDCOPY"
10 GRAPHIC 2,1,22
20 DRAW 1,0,2 TO 319,2: DRAW 1,0,4 TO 319,4: DRAW 1,0,
6 TO 319,6: BOX 1,0,0,319,199
30 CHAR 1,13,0,"SUPER-HARDCOPY",1
40 FOR I=1 TO 35: CIRCLE 1,30+I*7,80,15,60,,,I*10: NEX
T I
50 INPUT "X-VERGROESSERUNG ";X
60 INPUT "Y-VERGROESSERUNG ";Y
70 INPUT "DRUCKMODUS{7SPACE}";DR
80 SYS 4867,0,X*16+Y,DR
90 GOTO 50
```

Listing 20. Ein Demo zur Super-Hardcopy

21. VDC-Werte

Ich habe Ihnen hier eine Tabelle der sinnvollen Werte für die Register des VDC 0 zusammengestellt.

Beschrieben wird es mit SYS DEC("CDCC"),X,0.

X = 130 - 132 Der Bildschirminhalt wird nach oben gescrollt, bei kleineren Werten wird das Scrollen schneller.

X = 133 - 135 Der Bildschirminhalt wird nach unten gescrollt, bei größeren Werten wird das Scrollen schneller.

X = 243 - 255 Die Schrift wird in doppelter vertikaler Größe ausgegeben. Die exakte Steuerung der Bildschirmausgabe ist nicht mehr möglich.

Sobald andere Werte außer dem Normalwert und den oben angegebenen benutzt werden, wird der Bildschirm entweder abgeschaltet, oder es erscheint ein undefinierbares Flimmern, das nur mit <RUN-STOP/RESTORE> abgebrochen werden kann.

Ich möchte Ihnen noch ein Demo-Programm für das Scrolling geben.

```
10 A = DEC (''CDCC'')
20 SYS A,133,0   :REM Scrolling starten
30 GETKEY A$
40 SYS A,132,0   :REM In die erste Zeile zurückscrollen
```

Es können natürlich auch die oben aufgeführten Werte eingesetzt werden.

(Bernd Geyer/ef)

22. Komfortable Sprite-Steuerung

»Joysprite« (Listing 21) steuert Sprite 1 durch die Bewegung eines Joysticks in Port 2. So wird die Programmierung von Menüsteuerungen oder Malprogrammen deutlich vereinfacht.

Tippen Sie die Routine im C 64-Modus mit dem MSE ab. Im C128-Modus wird sie dann mit BLOAD geladen. Da Joysprite im Interrupt läuft, muß sie über den Vektor $314/$315 eingehängt werden. Diese Aufgabe sollte auch von Assembler-Neulingen recht schnell gelöst werden, zum Beispiel mit folgender Routine, die mit dem Monitor oder natürlich mit einem beliebigen Assembler eingegeben wird:

```
02000 SEI
02001 LDA #$0D
02003 STA $0314
02006 LDA #$20
02008 STA $0315
0200B CLI
0200C RTS
0200D JSR $1300
02010 JMP $FA65
```

Nach Eingabe dieser Zeilen und Aufruf mit Bank 15:SYS8192 wird das Sprite, das Sie beliebig definieren können, unabhängig von einem laufenden Basic-Programm gesteuert. Schalten Sie die Routine entweder durch <RUN/STOP RESTORE> aus oder durch Starten des etwas geänderten Initialisier-Programms. Dazu müssen nur zwei Zeilen einfach überschrieben werden:

```
02001 LDA #$65
```
und
```
02006 LDA #$FA
```
(K. H. Guckler/ef)

```
Name : joy-sprite 1.obj  1300 13a1
----------------------------------
1300 : ac 00 dc 98 29 01 d0 03  da
1308 : 20 2c 13 98 29 02 d0 03  02
1310 : 20 30 13 98 29 04 d0 03  1c
1318 : 20 61 13 98 29 08 d0 03  dd
1320 : 20 34 13 98 29 10 d0 03  8f
1328 : 20 90 13 60 ce d7 11 60  12
1330 : ee d7 11 60 ae d6 11 e8  12
1338 : f0 04 8e d6 11 60 ad e6  41
1340 : 11 29 01 f0 0e ad e6 11  50
1348 : 29 fe 8d e6 11 a2 00 8e  74
1350 : d6 11 60 ad e6 11 09 01  9a
1358 : 8d e6 11 a2 00 8e d6 11  e3
1360 : 60 ae d6 11 ca e0 ff f0  85
1368 : 04 8e d6 11 60 ad e6 11  bc
1370 : 29 01 f0 0e ad e6 11 29  c0
1378 : fe 8d e6 11 a2 ff 8e d6  2b
1380 : 11 60 ad e6 11 09 01 8d  82
1388 : e6 11 a2 ff 8e d6 11 60  44
1390 : ea ea ea ea ea ea ea ea  8f
1398 : ea ea ea ea ea ea 60 ff  97
13a0 : fb 20 d2 ff ee 52 15 ad  91
```

Listing 21. Einfache Sprite-Steuerung in Menüs

23. VDC-Window

Diese Routine erlaubt es, den Inhalt des 80-Zeichen-Bildschirms in den freien Speicher des VDC-Chips zu kopieren und von dort auch wieder zurückzuholen. Selbst der Attributspeicher wird hierbei berücksichtigt. Tippen Sie dazu Listing 22 mit dem MSE im C 64-Modus ab. In dem Beispiel-Programm (Listing 23) wird ein Bild erzeugt und dann in den Speicher kopiert. Jetzt kann dieses Bild von neuen Anweisungen geändert werden. Danach muß nicht wieder das alte Bild hergestellt werden, sondern es wird durch den Aufruf in Sekundenschnelle zurückgeholt.

Einsprungadresse
80-Zeichen-Bild-SAVE: $1303
80-Zeichen-Bild-LOAD: $1300 (K. H. Guckler/ef)

```
Name : vdc-window         1300 1391
-----------------------------------
1300 : 4c 55 13 a2 18 8e 00 d6  b4
1308 : 2c 00 d6 10 fb ad 01 d6  cb
1310 : 09 80 48 20 79 13 a0 10  42
1318 : a9 00 85 fa 85 fb 85 fd  cc
1320 : a9 10 85 fc a5 fc a2 12  c3
1328 : 20 79 13 a9 00 e8 20 79  ba
1330 : 13 a5 fa a2 20 20 79 13  38
1338 : a9 00 e8 20 79 13 a2 1e  16
1340 : a9 00 20 79 13 e6 fa e6  43
1348 : fc 88 d0 d8 68 29 7f a2  eb
1350 : 18 20 79 13 60 a2 18 8e  d2
1358 : 00 d6 2c 00 d6 10 fb ad  08
1360 : 01 d6 09 80 48 20 79 13  b0
1368 : a0 10 a9 00 85 fc 85 fd  cd
1370 : 85 fb a9 10 85 fa 4c 24  09
1378 : 13 8e 00 d6 2c 00 d6 10  eb
1380 : fb 8d 01 d6 60 00 00 00  63
1388 : 00 00 00 00 00 00 00 00  89
1390 : 00 ea ea ea ea ea ea ea  a5
```

Listing 22. »VDC-WINDOW« rettet den Fensterhintergrund

```
10 BANK 15: FAST : GRAPHIC 5,1
20 PRINT CHR$(19) CHR$(19) CHR$(27)"M";
30 FOR I=1 TO 2000
40 : IF RND(0)>.5 THEN PRINT CHR$(15);: ELSE PRINT CHR
   $(143);
50 :  COLOR 5,INT(RND(1)*15)+1
60 : PRINT CHR$(205+RND(0)+.5);
70 NEXT
80 PRINT "CHR$(19)CHR$(19)
90 :
100 SYS DEC("1303"): REM BILD SPEICHERN
110 :
120 PRINT CHR$(27)"L" CHR$(5) CHR$(143);
130 CHAR 1,30,12,"**********************"
140 CHAR 1,30,13,"*{20SPACE}*"
150 CHAR 1,30,14,"*{6SPACE}TASTE{9SPACE}*"
160 CHAR 1,30,15,"*{20SPACE}*"
170 CHAR 1,30,16,"* FUER ALTES BILD{4SPACE}*"
180 CHAR 1,30,17,"*{20SPACE}*"
190 CHAR 1,30,18,"*{20SPACE}*"
200 CHAR 1,30,19,"**********************"
210 GET KEY A$
220 :
230 SYS DEC("1300"): REM BILD WIEDERHERSTELLEN
240 :
250 SLEEP 1
260 GOTO 130
```

Listing 23. Ein Demo zu VDC-WINDOW

24. Defekt im Char-Befehl

Kürzlich habe ich bei meinem C 128 einen Fehler im Betriebssystem beziehungsweise Basic-ROM entdeckt, der zwar selten Auswirkungen zeigt, anderenfalls jedoch sehr unangenehm ist. Der Übeltäter ist der ach so schöne Befehl CHAR im Basic 7.0 des C 128. In der Wirkung funktioniert er zwar einwandfrei, doch was passiert wirklich, wenn man den CHAR-Befehl auf dem 80-Zeichen-Bildschirm anwendet? Das Basic-RAM wird beschrieben! Dies erfolgt an den Adressen $d600 und $d601 in RAM-Bank 0. Bei Basic-Programmen mit FRE(0)>10495 ist das nicht weiter tragisch, da diese Speicherstelle nicht erreicht wird. Bei sehr großen Programmen — und deshalb macht sich dieser Fehler so selten bemerkbar — wird das Basic-Programm verändert! Harmlos ist es noch, wenn sich an den Adressen $d600/d601 der Text zum Beispiel eines PRINT-Befehles befand. Hier wird dann meistens im Programmablauf etwas zu blinken anfangen. Schon schlimmer, wenn ein Token überschrieben wird. Dann führt das Programm einen falschen Befehl aus oder endet mit einer Fehlermeldung. Doch katastrophal ist es, wenn eine Basic-Zeilennummer überschrieben wurde. Der Rest des Programmes besteht dann nur noch aus »Nummernsalat« und es dürfte sich schwierig gestalten, es wieder zu rekonstruieren.

Geben Sie einmal auf dem 80-Zeichen-Schirm CHAR,9,9, "TEST" ein und schauen dann mit dem Monitor mit M 0d5f0 im Speicher der Bank 0 nach. Das Einschaltgewirr von »00«- und »ff«-Code ist jäh an den Adressen $d600/d601 durch »0f d9« verändert.

Doch nun hat es mir keine Ruhe gelassen. Adresse $d600/d601 und 80-Zeichen-Bildschirm? Da wird man doch stutzig. Das sind doch die Adressen des VDC-Chips. Irgendwo muß eine Adressierung an den VDC-Chip in Bank 0 rutschen. Schließlich tritt dieser Fehler beim 40-Zeichen-Schirm weder im Text- noch im Grafik-Modus auf.

Schuld ist hier ein Unterprogramm im Basic-ROM, welches der CHAR-Befehl an Adresse $f6838 aufruft. In diesem ab Adresse $f928d gelegenen Unterprogramm soll der

Cursor gesetzt werden. Das Fatale ist nun, daß hier als erstes mit dem Befehl sta $ff03 Bank 14 eingeschaltet wird (denn hier liegt das Load-Konfigurationsregister C und im Prä-Konfig.Reg. C an Adresse $fd503 ist nun normalerweise die Konfiguration $01 = Bank 14 eingestellt). Später wird die Position des Cursors in der eigentlichen Plot-Routine an den VDC übermittelt. Das sollte über die Adressen $d600/d601 in Bank 15 geschehen, also bei eingeschaltetem I/O-Bereich. Hier aber wird in Bank 14, also das Charakter-ROM, geschrieben, welches ja eigentlich zum Lesen gedacht ist. Die MMU verübelt nun dieses Unterfangen, indem sie kurzerhand in Bank 0 schreibt. Übrigens wird dann später im Verlauf der CHAR-Routine zur Ausgabe des Strings richtig durch

```
pha
lda#$00
sta$FF00
pla
```

auf den I/O-Bereich umgeschaltet.

Spätestens jetzt wird auch verständlich, welcher Code nun gerade in Bank 0 landet. Die Adresse des Cursors im VDC-Chip errechnet sich nach adresse=hex$(Zeile*80+Spalte). Bei CHAR,9,9 also — obiges Beispiel — gilt adresse=$02d9. Nun wird erst das High-Byte dieser Adresse in Register $0e=14 des VDC geschrieben. Dann das Low-Byte in Register $0f=15 des VDC. Deshalb steht nach CHAR,9,9 an Adresse $d600 ein $0f, also das Register des VDC, welches als letztes angesprochen wurde. An d601 steht dann das Low-Byte der Cursor-Adresse, also hier $d9. Versuchen wir zum Beispiel CHAR,0,3, so können wir mit dem Monitor überprüfen, daß folgerichtig »0f 03« an die Adressen $d600/d601 geschrieben wurde.

Nun ergeben sich zwei Fragen: 1. Wie kann man diesen Fehler umgehen? 2. Sind noch andere Befehle davon betroffen?

Zur ersten Frage: Noch einmal sei betont, daß man sich nur bei extrem langen Basic-Programmen Gedanken machen muß. Leicht zu überprüfen mit dem FRE-Befehl: Bei FRE(0)>10495 ist alles in Ordnung. Allerdings sollte vorher die Hires-Grafik schon einmal eingeschaltet worden sein, wenn das Programm mit Grafik arbeitet. — Ferner ist der CHAR-Befehl bedenkenlos auf dem 40-Zeichen-Bildschirm anzuwenden.

Um nun auch bei langen Programmen auf dem 80-Zeichen-Bildschirm den Cursor bequem zu positionieren, sind drei Lösungen denkbar. Die erste ist die Schlechteste. Man ändert durch

```
bank15:poke dec("d503"),0
```

das Präkonfigurationsregister C so ab, daß beim Beschreiben von ff03 immer Bank 15 eingeschaltet wird. Das wird aber oft in die Hose gehen, da nicht abzusehen ist, bei welchen Basic-Befehlen dann wiederum Fehler auftreten (bei einem ersten Versuch klappte es erstaunlich gut).

Die zweite Lösung bietet sich immer dann an, wenn man partout nicht auf den CHAR-Befehl verzichten möchte. Man rettet vor einem CHAR-Aufruf das Basic-RAM mit PEEK: am Anfang eines Programms:

```
D=DEC("D600"):BANK0:D1=PEEK(D):D2=PEEK(D+1)
```

bei jedem CHAR-Aufruf:

```
CHAR,.....:BANK0:POKED,D1:POKED+1,D2
```

Das geht immer gut, außer wenn zufällig der Schwanz eines CHAR-Befehles gerade an besagten Adressen d600/d601 liegt. Die dritte Lösung ist die sauberste, bedeutet jedoch mit dem Verzicht auf CHAR einen Rückschritt in alte C 64-Zeiten. Jeder 80-Zeichen-CHAR-Befehl wird durch ein SYS ersetzt:

```
BANK15:SYS DEC("FFF0"),,x,y:PRINT....
oder :SYS DEC("C018"),,x,y:PRINT....
```

wobei x=Spalte und y=Zeile die bekannten CHAR-Koordinaten bedeuten und der String im PRINT-Befehl ausgegeben wird.

Letzteres ist das probate (und wohl auch bekannte) Vorgehen in Maschinenprogrammen für die Positionierung des Cursors:

```
lda #$00
sta $ff00 ; Bank 15 eingeschaltet
...
ldx #Spalte
ldy #Zeile
clc
jsr $fff0 ; oder auch jsr $c018
```

Im x- beziehungsweise y-Register werden Spalte und Zeile übergeben, anschließend wird das eigentliche Cursor-Plot-Unterprogramm aufgerufen. Zuvor ist unbedingt das Carry-Flag zu löschen, wie im folgenden noch deutlich wird.

Denn nun zur zweiten Frage: Sind noch andere Befehle betroffen? In der Tat wird das Unterprogramm $928D insgesamt fünfmal aufgerufen (mit dem H-Befehl im Monitor ansehen). Um die Wirkungsweise dieser Aufrufe zu verstehen, sei kurz das Unterprogramm »Cursor-Plot« skizziert. Es ist - wie beim C 128 üblich - ein Universalgenie. Zunächst wird bei $928D auf Bank 14 umgeschaltet, wie oben beschrieben. Dann folgt die eigentliche Plot-Routine. Diese erfüllt aber gleich zwei Aufgaben: Ist das Carry-Flag gelöscht, wird der Cursor entsprechend x- und y-Register gesetzt. Ist das Carry-Flag dagegen gesetzt, so wird die aktuelle Cursorposition gelesen und im x- beziehungsweise y-Register übermittelt. Ferner wird jedesmal unterschieden, welcher Bildschirm-Modus gerade aktiv ist. Jetzt lassen sich die Konsequenzen im einzelnen untersuchen (siehe Tabelle 2):

Aufruf bei	betrifft Befehle	Auswirkungen
F55AA und F55BB	SPC,TAB	keine! Hier wird die Cursorposition nur gelesen und dabei ausschließlich auf Zeropage zugegriffen, nicht auf den VDC.
F6838	CHAR	Fehler! Wie beschrieben Zugriff auf VDC.
F6ABF	SCNCLR	Keine! Hier wird der Cursor zwar gesetzt, aber mit jsr$928D nur auf dem 40-Zeichen-Bildschirm.
F84D1	POS	Keine! Cursor wird nur gelesen (siehe SPC,TAB).

Tabelle 2. Aufrufe des Unterprogrammes ab 928D in Basic 7.0

Somit ist das Überschreiben von Basic-Programmen also wirklich nur bei CHAR zu befürchten! (T. Sperling/ef)

25. Der VDC des Blech-Diesels

Der VDC ist wohl der interessanteste Baustein im C 128. Bei längerem Arbeiten mit diesem gerüchteumwitterten Videoprozessor wurde versehentlich das Register 37 beschrieben, das — laut Commodore-Handbuch — nicht existiert. Die Wirkung, die durch das Beschreiben entsteht, ist für eigene Zwecke kaum zu gebrauchen, läßt sich aber dennoch sehen. Von diesem Register sind nur die Bits 6 und 7 benutzt. Bit 6 bewirkt, daß der Bildschirm an den rechten Rand gerückt wird. Bit 7 erzeugt ein durchlaufendes Bild. Interessant ist noch, daß ein Zurücksetzen des Registers

auf den Normalwert nur durch ein Beschreiben mit 192 bis 255 oder einen Kaltstart (Reset-Taster) möglich ist. Die Tastenkombination <RUN/STOP RESTORE> ändert dieses Register genausowenig wie der Software-Reset SYS DEC("FF3D").

Und wieso? Weil das Register vom Betriebssystem des C128 her nicht unterstützt wird. (T. Schadde/ef)

26. Rasterzeilen flimmerfrei

Viele Programmierer, sicherlich auch viele Anwender, stört das relativ starke Flimmern zwischen den zwei Bildschirmbereichen nach einem GRAPHIC2- oder GRAPHIC4-Befehl. Um dieses Flackern beim Rasterzeileninterrupt zu verhindern, muß die Rasterzeile, bei der ein Interrupt ausgelöst wird, um 1 verringert werden. Dazu sollte man wissen, daß der Computer in der Speicherstelle 2512 (in der erweiterten Zeropage) eben genau diese Rasterzeile speichert. Nach jedem Aufruf von GRAPHIC2 oder GRAPHIC4 wird dort diese Rasterzeile abgelegt. Wir brauchen nur von diesem Wert 1 abziehen, und das Flackern hört auf. Wenn das geschehen ist, werden Sie feststellen, daß in der untersten Rasterzeile ein farbiger Streifen erscheint. Dieser wird durch das Zeichen-RAM der Grafik erzeugt, das auch weiterhin nur eine Farbe je volle acht Rasterzeilen darstellt. Verhindern läßt sich dieser Streifen, indem man die letzten acht Rasterzeilen der Grafik nicht verwendet, oder diese, wie im Demoprogramm (Listing 24), mit Spaces überschreibt. Wer mit dieser Lösung nicht zurechtkommt, der wird wohl nicht umhinkommen, eine komplett neue Interruptroutine zu schreiben. Nur dann ist ein uneingeschränkter, flackerfreier Rasterzeileninterrupt möglich.

(D. Bozza/ef)

```
10 REM ******************
20 REM *   DIRK BOZZA   *
30 REM * GRUENER WEG 12 *
40 REM *  5800 HAGEN 5  *
50 REM ******************
60 REM
70 COLOR 0,1: COLOR 1,15: COLOR 4,1
80 GRAPHIC 2,1: COLOR 0,7
90 PRINT "{CLR,16DOWN,YELLOW,3SPACE}RASTERZEILENINTERR
   UPT MIT FLIMMERN"
100 PRINT "{2SPACE}*************************************
    *"
110 GRAPHIC 2,0,14
120 BOX 1,0,0,319,103
130 BOX 1,5,5,314,98
140 PAINT ,1,1
150 CHAR 1,5,6,"BITTE DRUECKEN SIE EINE TASTE!"
160 GET KEY A$
170 PRINT "{CLR,16DOWN,3SPACE}RASTERZEILENINTERRUPT OH
    NE FLIMMERN"
180 PRINT "{2SPACE}*************************************
    **"
190 REM
200 REM ************************
210 REM * FLIMMERN AUSSCHALTEN *
220 REM ************************
230 REM
240 GRAPHIC 2,0,13
250 X=7168+((PEEK(2612)-48)/8)*40
260 POKE 2612,PEEK(2612)+7
270 FOR M=0 TO 39: POKE X+M,32: NEXT
280 REM
290 REM ************************
300 REM
310 COLOR 1,3
320 CHAR 1,4,2,"ZU BEACHTEN IST, DAS DIE LETZTEN"
330 CHAR 1,7,4,"8{2SPACE}RASTERZEILEN DER GRAFIK"
340 CHAR 1,5,6," NICHT BENUTZT WERDEN KOENNEN "
350 COLOR 1,8: FOR X=100 TO 220 STEP 10
360 CIRCLE 1,X,82,10,10: NEXT
370 COLOR 1,2: CHAR 1,12,10,"TASTE FUER ENDE!"
380 GET KEY A$: GRAPHIC 0
```

Listing 24. Flackerfrei mit Rasterzeile © 64'er

27. Vereinfachtes Laden

Obwohl im Basic V7.0 des C 128 die Befehle DLOAD und BLOAD implementiert sind, ist es doch umständlich, Programme aus einem Directory zu laden.

Zwar muß man nicht mehr hinter dem Programmnamen das gefürchtete »,8« anhängen, aber da der Computer nicht die Dateiart ignoriert (wie bei diversen Erweiterungen üblich), ist man gezwungen, diese Kennung mittels Cursor zu löschen. Es geht aber auch anders: Setzen wir doch nach dem Programmnamen einen Doppelpunkt. Es ist überhaupt nicht schwer, im Directory beliebige Zeichen hinter den Dateinamen zu setzen. Der Computer füllt nämlich den Rest eines Namens mit geSHIFTeten <SPACE> auf. Alle Zeichen, die nach dem ersten Space stehen, gehören nicht mehr zum Namen und werden ignoriert. Man muß nur darauf achten, daß Programmname, <SHIFT SPACE> und die folgenden Zeichen insgesamt nur 16 Zeichen lang sind. Wir wollen nur einen Doppelpunkt hinter einen Namen bringen, so daß wir zwei Zeichen nach dem Namen benötigen. Insgesamt darf der Name also 14 Zeichen lang sein.

Nachdem man den Programmnamen zum Speichern eingetippt hat, gibt man (vor dem zweiten Anführungszeichen) ein <SHIFT SPACE> und einen Doppelpunkt ein. Im Directory stehen dann anstelle des <SPACE> die zweiten Anführungszeichen und danach der Doppelpunkt. Nun muß man zum Laden nur noch auf den Anfang der entsprechenden Zeile fahren, »DLOAD«, »RUN« oder <F2> eingeben und das Programm wird anstandslos geladen. Übrigens reicht es, nach »DLOAD« oder »RUN« den tatsächlichen Namen anzugeben. Allerdings werden beim Laden von Maschinensprache-Programmen mittels »BLOAD« keine Zeiger im Computer verändert, so daß er den folgenden (vermeintlichen) Befehl auch noch ausführen will — ein »SYNTAX ERROR« wäre die Folge. Um dies zu verhindern, schreibt man nach dem Doppelpunkt noch ein REM, was zwar etwas irritiert, aber diese Meldung verhindert.

(T. Gudella/ef)

28. Neues von der 1571

Im Floppy-Betriebssystem ist noch ein bisher nicht erwähnter Fehler vorhanden: Wird beim Speichern der letzte Block eines Programms vollständig gefüllt, dann wird ein Block zuviel belegt. Im Directory erfolgt der Eintrag korrekt, allerdings »fehlt« auf der Diskette ein Block. Ein Validate stellt hier jedoch wieder Ordnung her. Dieser Fehler ist bei meiner 1571 (Blech) in allen Betriebsarten (auch bei Dolphin-DOS) vorhanden. Hier ein Beispiel: Geben Sie vom Monitor aus »s"name",8,1c01,1cfd« oder »s"name",8,1c01,1dfb« ein. Beim ersten Speichern wird ein kompletter Block gebraucht, beim zweiten zwei. Der Eintrag der Blocks erfolgt korrekt mit 1 beziehungsweise 2, es werden jedoch 2 beziehungsweise 3 Blocks belegt. Erst ein VALIDATE korrigiert die Eintragungen. Solange man nicht mit Direktzugriffsdateien arbeitet, ist dieser Fehler bedeutungslos. Peinlich wird es jedoch, wenn auf diese Weise ständig Speicherplatz »verdunstet« und ein Validate nicht erlaubt ist. Scratch-geschützte Programme können übrigens mit der 1571 (Blech) gelesen werden. Der in der Ausgabe 12/87 erschienene DOS-Patch ist hier nicht nötig und wahrscheinlich auch nicht funktionsfähig, da im DOS der 1571 (Blech) ja einige Veränderungen vorgenommen wurden. Der Block-Allocate- und der Block-Free-Befehl funktionieren übrigens überhaupt nicht. Versuchen Sie einmal, auf einer neu formatierten Diskette Block 1, Sektor 0 zu belegen.

Spätestens nach dem nächsten Initialize zeigt sich die Wahrheit. Die komplette Spur 1 ist belegt! Auch ein Block-Free hilft nicht. Auf der Diskette zumindest ist und bleibt die Spur belegt. Es bleibt also nichts weiter übrig, als direkt die BAM zu manipulieren (wie mit dem Autoboot Maker von der 1571-Test-Diskette). Beim Dolphin-DOS stört ein Schönheitsfehler den Betrieb mit der 1571. Validierte man bisher eine zweiseitige Diskette versehentlich im 1541-Modus, so fehlten anschließend 664 Blocks. Dies konnte jedoch durch ein Validate im 1571-Modus wieder behoben werden. Nicht so bei Dolphin-DOS! Hier hilft nur noch ein Disk-Monitor, um die Kennung für eine zweiseitige Diskette wiederherzustellen. (T.Lang/ef)

29. Noch einmal 1571

40-Spur-Disketten lassen sich mit der 1571 einfach lesen. Geben Sie dazu bitte folgende Befehle im Direktmodus ein:

```
open1,8,15,»u0>m0«
print#1,»m-w«chr$(172)chr$(2)chr$(1)chr$(41)
close 1
```

Noch etwas: Vorsicht ist geboten, wenn man den C 128D (Blech) offen betreibt (Wärmeabfuhr). Da die Schreibschutz-Lichtschranke der Floppy hier offen liegt, werden Schreibschutzaufkleber wirkungslos, sobald genug Fremdlicht auf die Lichtschranke fällt. Ich selbst habe mir damit eine meiner wichtigsten Disketten zerstört!

Diese Eigenschaft läßt sich aber auch sinnvoll einsetzen. Wenn der Computer sowieso offensteht, braucht man bloß eine Taschenlampe über die Lichtschranke zu stellen und kann dann auch die Rückseite einer Diskette beschreiben, ohne diese »anzuschneiden«. So spart man sich den Schreibschutzaufkleber (das sind die Dinger, die immer dann im Laufwerk steckenbleiben, wenn man es am wenigsten erwartet).

Auf der zweiten Seite kann sich dann zum Beispiel die Programmdiskette befinden, auf der ungeschützten ersten Seite die Daten. Vor einem Diskettenwechsel muß allerdings das Fremdlicht abgeschaltet werden, damit die neue Diskette initialisiert wird. (T. Lang/ef)

30. Kopierschutz einmal einfach

Ein einfacher, aber wirkungsvoller Kopierschutz, der allerdings nur mit der 1571 im 1541-Modus funktioniert, ist das Speichern von Programmen auf der zweiten Diskettenseite. Auf der ersten Seite befindet sich ein Lader, der denselben Namen wie das zu ladende Programm haben kann. Er ist einen Block lang und besteht aus nur einer Basic-Zeile:

```
0 OPEN 1,8,15,"U0>H1":LOAD "HAUPTPROGRAMM",8
```

Der erste Befehl stellt die Floppy auf den zweiten Kopf um. Dann wird von der zweiten Seite das Programm »Hauptprogramm« geladen und automatisch gestartet. Sämtliche mir bekannten Kopierprogramme für die 1571 (egal ob 1541 oder 1571-Modus) können die zweite Seite nicht kopieren. Thirty Seconds gibt Lesefehler aus, da die Spuren 36 bis 70 erwartet werden. Tornado-Copy erlaubt keine Umstellung auf die zweite Seite, da es im 1571-Modus läuft. Mastercopy läuft nur mit der 1541, allerdings sind dann beim Wenden der Diskette die Daten in der falschen Richtung aufgezeichnet. File-copy-Programme lesen nur den Lader auf der ersten Seite und kopieren diesen. Kopieren ist also nur durch Einlesen und wieder Speichern möglich. Bei verschiedenen Files wird das aber etwas schwierig (REL, USR). (D. Astrath/ef)

31. Kopieren mit Laufwerk 9

Sehr viele Kopierprogramme funktionieren aufgrund der anderen Diskettenstation nicht auf dem neuen C128 D (Blech). Besitzer einer Zweitfloppy (1570/1571) sind ab jetzt in der Lage, Disketten schnell zu kopieren. Dazu wird die Geräteadresse eines Kopierprogramms nach dem Laden auf 9 gestellt. Besitzer von TORNADO COPY (64'er-Sonderheft 10) haben nur zwei POKE-Befehle einzugeben:

```
POKE 7905,9
POKE 7932,9
```

Anwender des Programms THIRTY SECONDS (64'er-Sonderheft 15) haben etwas mehr zu tun:

```
POKE 7375,9
POKE 7393,9
POKE 7463,9
POKE 8281,9
POKE 8301,9
```

Nach der Eingabe der POKEs sollte man das Programm speichern. Anstatt der 9 läßt sich auch eine andere Geräteadresse von 8 bis 12 eingeben. (U. Kepper/ef)

32. Hilfe zum MSE

Mit dem Listing 25 »MSE-Tipphilfe« erzeugen Sie auf Ihrem C 128 im C 64-Modus einen hexadezimalen Ziffernblock. Die Tipphilfe belegt die vier Funktionstasten mit den Buchstaben »A« bis »D«, < + > und <-> mit »E« und »F«. Der übrige Ziffernblock ist wie im C 128-Modus belegt.

Die »MSE-Tipphilfe« baut auf dem im 64'er-Sonderheft 1/86, Seite 142, veröffentlichten Programm »Key 128« auf. Es wurde um eine Funktionstasten-Abfrage erweitert und soweit gekürzt, daß es in den Kassettenpuffer paßt. Deshalb konnten auch der Interrupt- und der Tastaturdecodier-Vektor nicht mehr innerhalb des Assemblerprogrammes auf die erweiterten Routinen umgestellt werden. Dies geschieht durch die Zeilen 190 und 200 des Basic-Laders. Die beiden POKE-Kommandos in den Zeilen 180 und 210 entsprechen den Assembler-Befehlen SEI und CLI. Sie verhindern, daß das Betriebssystem in die Interrupt-Routine einspringt, bevor der IRQ-Vektor komplett umgestellt wurde.

Laden Sie zunächst das Programm mit »LOAD"MSE-TIPPHILFE",8«, starten es mit RUN und aktivieren dann den MSE ganz normal mit »LOAD"MSE",8« und RUN. Die Geschwindigkeit, mit der sich Maschinensprache-Programme eingeben lassen, ist nun beträchtlich höher als bisher.

Zwei Punkte sind noch zu beachten: Durch Drücken von <RUN/STOP RESTORE> deaktivieren Sie die MSE-Tipphilfe. Außerdem verträgt sie sich nicht mit Programmen, die ebenfalls im Kassettenpuffer abgelegt sind. Sie erkennen diese Programme an der Startadresse in der Kopfzeile des Listings. Diese Adresse darf nicht im Bereich von $033C bis $0400 liegen. (C. Högl/ef)

```
100 REM --- MSE-TIPPHILFE AM 128ER ---        <238>
110 REM ----------------------------          <142>
120 FOR L=820 TO 1019                         <152>
130 READ D : POKE L,D : P=P+D                 <121>
140 NEXT                                      <150>
150 :                                         <126>
160 IF P<>21212 THEN PRINT"ERROR":STOP        <212>
170 :                                         <146>
180 POKE 56334, 0: REM   INTERRUPT AUS        <166>
190 POKE 655, 166: POKE 656, 3                <142>
```

Listing 25. Die »MSE-Tipphilfe« hilft bei MSE-Listings

```
200 POKE 788,  52: POKE 789, 3                      <025>
210 POKE 56334, 1: REM   INTERRUPT AN               <091>
220 END                                             <222>
230 :                                               <206>
240 DATA 169, 64,133,203, 41,  0,141, 47            <037>
250 DATA 208,  9,255,141,  0,220,205,  1            <209>
260 DATA 220,240, 66,160, 89,169,251,141            <242>
270 DATA  47,208, 72,173,  1,220,205,  1            <149>
280 DATA 220,208,248,162,  8, 10,176,  2            <228>
290 DATA 132,203,136,192, 65,144,  7,202            <125>
300 DATA 208,243,104, 74, 16,225,104,173            <105>
310 DATA 141,  2, 10,201,  8,144,  2,169            <118>
320 DATA   6,170,189,188,  3,133,245,189            <032>
330 DATA 189,  3,133,246, 32,155,  3, 32            <019>
340 DATA 224,234, 32,155,  3,169,255,141            <225>
350 DATA  47,208, 41,127,141,  0,220,164            <042>
360 DATA 203,140,196,  3, 76, 49,234,165            <029>
370 DATA 197,174,196,  3,141,196,  3,134            <184>
380 DATA 197, 96,165,203,162,  3,221,244            <175>
390 DATA   3,240,  6,202, 16,248, 76, 72            <106>
400 DATA 235,189,248,  3,133,203,208,246            <103>
410 DATA 131,  3,155,  3,155,  3,131,  3            <204>
420 DATA  64,  8, 56, 53,142, 50, 52, 55            <241>
430 DATA  49,  0, 69, 70, 17, 13, 54, 57            <137>
440 DATA  51, 14, 48, 44,145, 17,157, 29            <133>
450 DATA   3,  9, 56, 53, 24, 50, 52, 55            <156>
460 DATA  49, 27, 43, 45, 10,141, 54, 57            <103>
470 DATA  51,142, 48, 58,145, 17,157, 29            <021>
480 DATA   3,  6,  5,  4, 18, 20, 28, 10            <025>
© 64'er
```

Listing 25. »MSE TIPPHILFE« (Schluß)

33. Fastload für C 64-Modus

90 Prozent aller Programme für den C 128 laufen immer noch im C 64-Modus. Es liegt nahe, die erhöhte Ladegeschwindigkeit der 1570/71 auszunutzen — genau hier setzt »BOOT 64« an. Geben Sie dazu bitte »BOOT64 INSTALL« (Listing 26) im C 128-Modus und Listing 27 im C 64-Modus mit dem MSE ein und speichern sie auf eine freie Diskette. Laden Sie jetzt »BOOT64« im C 128-Modus mit DLOAD, geben POKE 7309,147 ein und speichern es mit DSAVE ″BOOT64H″.

Das Installationsprogramm wird mit RUN″BOOT64 INSTALL″ gestartet. Jetzt entscheiden Sie sich entweder für eine Boot-(»AUTOSTART«) oder für eine im Direkt-Modus startbare Version (»HANDBETRIEB«). Für die Autostart-Version muß später mit dem »AUTOBOOT MAKER« von der »1570/71 Test/Demo Disk« ein entsprechender Boot-Sektor (für eine Basic-Datei) eingerichtet werden. Nach Ihrer Auswahl erscheint die Aufforderung, den Namen des C 64-Programms und der neu anzulegenden Start-Datei einzugeben. Jetzt erst legen Sie die C 64-Diskette (mindestens »2 BLOCKS FREE«) ein und drücken <SPACE> — fertig! In Zukunft wird der Lader Ihres Programms nur mit RUN ″name″ im C 128-Modus gestartet. (T. Lang/ef)

```
100 CLR
110 PRINT "{CLR}BOOT64 INSTALLATION": PRINT "{DOWN}VER
    SION FUER {RVSON}H{RVOFF}ANDBETRIEB / {RVSON}A{RVO
    FF}UTOSTART{DOWN}"
120 DO
130 GET KEY AW$
140 LOOP UNTIL AW$="A" OR AW$="H"
150 IF AW$="A" THEN AW$=""
160 DIM A$(1)
170 OPEN 1,8,2,"BOOT64"+AW$+",P,R": I1=1
180 IF DS THEN PRINT "FEHLER !{DOWN}": PRINT DS$: END
190 DO
200 : GET #1,A$: A$(I2)=A$(I2)+LEFT$(A$+CHR$(0),1)
210 : I1=((I1+1) AND 255): I2=I2-(I1=0)
220 LOOP UNTIL (ST AND 64)=64
```

Listing 26. Das Installationsprogramm für Listing 27

```
230 CLOSE 1
240 NA$=MID$(A$(0),44,17): FOR I=1 TO 17: IF MID$(NA$,
    I,1)<>CHR$(0) THEN NEXT I
250 NA$=MID$(NA$,1,I-1)
260 PRINT "BOOT-PROGRAMM{3SPACE}"NA$;LEFT$("{23SPACE}"
    ,23-LEN(NA$));
270 PRINT "{25LEFT}";
280 INPUT N$
290 IF LEN(N$)>16 THEN BEGIN
300 : PRINT : PRINT "NAME ZU LANG !{3UP}"
310 : GOTO 260
320 : BEND
330 FOR A=0 TO 16
340 N$=N$+CHR$(0)
350 NEXT A
360 N$=LEFT$(N$,17)
370 PRINT : PRINT "{14SPACE,2UP}"
380 PRINT "LADE-PROGRAMM{3SPACE}BOOT64{17SPACE,25LEFT}
    ";
390 INPUT NP$
400 IF LEN(NP$)>16 THEN BEGIN
410 : PRINT : PRINT "NAME ZU LANG !{3UP}"
420 : GOTO 380
430 : BEND
440 PRINT : PRINT "{14SPACE}": PRINT "ZIELDISKETTE EIN
    LEGEN": PRINT : PRINT "WEITER MIT <SPACE>"
450 DO
460 : POKE 208,0: WAIT 208,1: GET AW$
470 LOOP UNTIL AW$=" "
480 OPEN 1,8,2,(NP$)+",P,W"
490 IF DS=63 THEN BEGIN
500 CLOSE 1
510 PRINT "FILE EXISTIERT BEREITS !": PRINT "UEBERSCHR
    EIBEN ? (J/N)"
520 DO
530 POKE 208,0: WAIT 208,1: GET AW$
540 LOOP WHILE AW$=" "
550 IF AW$<>"J" THEN PRINT "ENDE": END
560 SCRATCH(NP$): OPEN 1,8,2,(NP$)+",P,W"
570 BEND
580 IF DS=>20 THEN PRINT "FEHLER !{DOWN}": PRINT DS$:
    PRINT "{DOWN}ENDE": END
590 PRINT#1,MID$(A$(0),1,43)N$ MID$(A$(0),61,LEN(A$(0)
    )-60)A$(1);
600 CLOSE 1
610 PRINT "FERTIG !"
```

Listing 26. Das Installationsprogramm (Schluß)

```
Name : boot64             1c01 1d27
-----------------------------------
1c01 : 28 1c c3 07 9e 20 28 37  03
1c09 : 32 32 37 29 3a 8f 14 14  e0
1c11 : 14 14 20 fe 1b 36 34 20  8c
1c19 : 42 59 20 a4 52 53 54 45  40
1c21 : 4e 20 4c 41 4e 47 00 00  da
1c29 : 00 2a 00 00 00 00 00 00  3f
1c31 : 00 00 00 00 00 00 00 00  32
1c39 : 00 00 a2 0c bd a8 1c 9d  30
1c41 : 00 13 ca 10 f7 a2 70 bd  51
1c49 : b4 1c 9d 00 cf ca 10 f7  f6
1c51 : 78 a9 85 8d 4c 03 a2 00  18
1c59 : 86 d2 a2 1a 8e d1 00 a2  f1
1c61 : 03 bd 8d 1c 9d 0a 10 ca  2a
1c69 : 10 f7 a2 ff e8 ee d1 00  6b
1c71 : ee 82 1c bd 2a 1c 9d 0e  75
1c79 : 10 d0 f1 a2 16 bd 91 1c  90
1c81 : 9d 0d 10 ca 10 f7 a9 0b  80
1c89 : 85 f1 58 60 97 42 cc 22  2c
1c91 : 22 2c 50 37 31 36 39 0d  88
1c99 : 42 41 4e 4b 31 35 3a 53  c5
1ca1 : 59 53 34 38 36 34 0d 78  e2
1ca9 : ad 05 d5 09 40 8d 05 d5  a0
1cb1 : 4c 00 cf a5 01 29 fe 85  06
1cb9 : 01 a2 00 bd 00 1c 9d 00  1a
1cc1 : 08 e8 d0 f7 ee 0a cf ee  cd
1cc9 : 0d cf ad 0a cf c9 ce d0  93
1cd1 : ea a5 01 09 01 85 01 38  a0
1cd9 : a5 ae 8d 70 cf a5 af e9  03
```

Listing 27. »BOOT 64« lädt C 64-Programme mit den schnellen Routinen des C 128

```
1ce1 : 14 8d 71 cf a2 ff 78 9a  53
1ce9 : d8 a2 04 8e 16 d0 20 a3  95
1cf1 : fd 20 50 fd 20 15 fd 20  b5
1cf9 : 5b ff 58 20 53 e4 20 bf  ca
1d01 : e3 ad 70 cf 85 2d ad 71  2c
1d09 : cf 85 2e a2 08 86 c6 bd  c6
1d11 : 67 cf 9d 76 02 ca d0 f7  40
1d19 : 4c 9d e3 43 4c 52 3a 52  7a
1d21 : 55 4e 0d 00 00 20 70 70  84
```

Listing 27. »BOOT 64« (Schluß)

34. Tipphilfe

»Tipphilfe« (Listing 28) hilft, C 64-Programme im C128-Modus (mit allen Eingabehilfen des Basic 7.0) zu editieren. Geben Sie das Listing bitte mit dem MSE ein und speichern es auf eine freie Diskette. Bevor Sie mit der Eingabe beginnen, laden Sie das Programm bitte mit »BLOAD 'TIPPHILFE', ONB0« (im C 128-Modus). Nachdem sich der Computer mit »READY.« gemeldet hat, können Sie beginnen, das C 64-Programm zu editieren. Nach vollständiger Eingabe des Programms geben Sie folgende Befehlssequenz im Direktmodus (!) ein: BANK 15: SYS DEC (»1300«). Der Computer schaltet sich in den 64er-Modus und listet Ihr Programm zur Kontrolle noch einmal auf. Wollen Sie jedoch, daß Ihr Programm sofort gestartet wird, ändern Sie die Bytes ab Adresse 1380 wie folgt: 1380: A9 52 A2 75 A0 0D 8D 77 41.

(M. Papke/ef)

```
Name : tipphilfe            1300 13a1
-----------------------------------
1300 : a9 f7 8d 05 d5 a2 ff 78  0c
1308 : 9a d8 e8 8e 16 d0 20 a3  ca
1310 : fd 20 50 fd 20 15 fd a9  e7
1318 : 03 85 9a a9 00 85 99 20  8c
1320 : 5b ff 20 02 fd 20 1b e5  dc
1328 : 20 53 e4 20 bf e3 20 22  0f
1330 : e4 78 a9 e0 85 01 a0 00  ba
1338 : b9 46 13 99 3c 03 c8 c0  8d
1340 : 60 90 f5 4c 3c 03 a2 00  56
1348 : 8a a8 a9 01 85 f7 a9 1c  a8
1350 : 85 f8 a9 01 85 f9 a9 08  bb
1358 : 85 fa b1 f7 91 f9 c8 d0  73
1360 : f9 a0 00 e6 f8 e6 fa e8  0b
1368 : e0 dc 90 ee a9 e7 85 01  aa
1370 : a9 08 8d 02 08 20 33 a5  5b
1378 : a5 22 a6 23 85 2d 86 2e  75
1380 : a9 52 a2 75 a0 0d 8d 77  41
1388 : 02 8e 78 02 8c 79 02 a9  20
1390 : 03 85 c6 a2 81 6c 00 03  dd
1398 : 60 00 00 00 00 00 00 00  f9
13a0 : 00 20 d2 ff ee 52 15 ad  96
```

Listing 28. »TIPPHILFE«. Geben Sie C 64-Programme im C 128-Modus ein.

35. Basic 2.0 Checksummer

In Tip 34 haben wir bereits eine Eingabehilfe für C 64-Basic-Programme veröffentlicht. Die Erweiterung brachte den Vorteil mit sich, daß Basic 2.0-Programme mit den komfortablen Programmierhilfen des Basic 7.0 eingegeben werden konnten. Ein Abtippen mit dem Checksummer schied bisher allerdings aus. Abhilfe schafft hier der Checksummer C 64-C 128 (Listing 29), der (für Basic 2.0-Befehle) die gleichen Prüfsummen benutzt wie das C 64-Äquivalent. Somit geben Sie nun Ihre C 64-Listings unter der gewohnten Checksummer-Umgebung, aber mit den leistungsfähigen Hilfen des Basic 7.0 im C 128-Modus ein.

Eingabehinweise: Tippen Sie Listing 29 mit dem MSE im C 64-Modus ab. Nach dem Ladevorgang mit BLOAD im C 128-Modus aktivieren Sie den Checksummer C 64-C 128 mit SYS DEC("C03"). Ein Start mit BOOT ist nicht möglich.

(D. Astrath/ef)

```
Name : checksummer 128   0c03 0c89
-----------------------------------
0c03 : a9 0c 8d 02 03 8d 03 03  05
0c0b : 60 38 a5 61 e9 02 85 61  a4
0c13 : b0 02 c6 62 a9 3f 8d 00  8d
0c1b : ff a9 00 a8 aa 85 fe b1  3a
0c23 : 61 f0 28 c9 20 d0 03 c8  66
0c2b : d0 f5 85 ff 8a 29 07 aa  ba
0c33 : f0 0e 48 a5 ff 18 2a 69  2d
0c3b : 00 ca d0 f9 85 ff 68 aa  63
0c43 : e8 a5 ff 18 65 fe 85 fe  63
0c4b : 18 90 dc c0 02 30 db a9  5f
0c53 : 00 8d 00 ff c6 eb a5 eb  54
0c5b : 48 20 7d ff 12 13 3c 00  bd
0c63 : a6 fe a9 00 20 32 8e 20  01
0c6b : 7d ff 3e 1b 51 0d 00 68  29
0c73 : 85 eb a9 8d 20 d2 ff 18  d3
0c7b : a5 61 69 02 85 61 90 02  15
0c83 : e6 62 4c c6 4d 31 30 30  06
```

Listing 29. »CHECK C 64-C 128« hilft bei der Eingabe von Basic-Listings für den C 64.

36. C 64 oder C 128

Mir ist aufgefallen, daß im C 64-Modus des C128, im Gegensatz zum »echten C64«, das Bit 6 des Prozessorports (Speicherstelle »1«) immer gesetzt ist. Dadurch ergibt sich eine einfache Abfrage, die den C 64 vom C 128 im C 64-Modus unterscheidet. Eine entsprechende Routine könnte etwa so aussehen:

```
10 POKE 1, PEEK(1) AND 255-64
11 REM BIT 6 ZURUECKSETZEN
20 A=PEEK(1) AND 64
21 REM BIT 6 TESTEN
25 :
30 IF A=0 THEN PRINT"RICHTIGER C 64":END
40 PRINT"C 128 IM C 64-MODUS"
```

Nun kann ein eigenes Programm den Computertyp feststellen, um gegebenenfalls im C 64-Modus zwischen ein und zwei Megahertz Taktfrequenz hin- und herzuschalten. Der neue C 128D im Blechgehäuse ist von diesem Phänomen übrigens genauso wie seine Vorgänger betroffen.

(H. Stöcklein/ef)

37. C128 gleich C 64???

Doch wie so viele Dinge auf dieser Welt haben einfache Lösungen oft einen Haken:

Wenn bei einem C 128 die Taste ASCII/DIN gedrückt ist, wird Bit 6 des Prozessor-Ports gelöscht. In diesem Fall besitzt die Speicherzelle »1« eines C 128 im C 64-Modus exakt den gleichen Wert wie die eines »echten C 64«. Daher ist die Abfrage des Computer-Typs durch dieses Register nur im ASCII-Modus sicher.

(M. Hensel/ef)

38. Benutzung der Zusatztasten im C 64-Modus

Die Benutzung der Zusatztasten (10er-Block usw.) ist auch im C 64-Modus möglich. Man lädt einfach (im C 64-Modus) das Programm »ZUSATZTASTEN« (Listing 30) mit:
LOAD"ZUSATZTASTEN",8 <RETURN>
und startet es mit RUN. Nach zirka drei Sekunden meldet sich der Computer wieder mit READY. Nun sind die zusätzlichen Tasten »aktiviert« (siehe Tabelle 3).

```
10 AN=828:REM ANFANGSADRESSE                   <153>
20 FOR I=0 TO 188:READ B:POKE AN+I,B:NEXT
   I                                           <234>
30 POKE AN+157,PEEK(788):POKE AN+158,PEEK(
   789):REM NORMALER IRQ                       <050>
40 HI=INT((AN+165)/256):LO=(AN+165)-256*HI     <139>
50 POKE AN+147,LO:POKE AN+148,HI:REM ANFAN
   G ASCII TABELLE                             <000>
60 POKE 56334,PEEK(56334)AND 254:REM IRQ A
   US                                          <125>
70 HI=INT(AN/256):LO=AN-256*HI                 <134>
80 POKE 788,LO:POKE 789,HI:REM NEUER IRQ       <236>
90 POKE 56334,PEEK(56334)OR 1:REM IRQ WIED
   ER AN                                       <106>
100 END                                        <102>
110 DATA 120,165,198,201,10,240,27,169,255
    ,141,0,220,173,47,208                      <112>
120 DATA 41,248,141,47,208,172,1,220,9,7,1
    41,47,208,192,255                          <143>
130 DATA 240,127,208,2,240,120,160,0,173,4
    7,208,41,254,141,47                        <185>
140 DATA 208,174,1,220,9,7,141,47,208,138,
    201,255,208,46,160                         <209>
150 DATA 8,173,47,208,41,253,141,47,208,17
    4,1,220,9,7,141                            <205>
160 DATA 47,208,138,201,255,208,23,160,16,
    173,47,208,41,251,141                      <085>
170 DATA 47,208,174,1,220,9,7,141,47,208,1
    38,201,255,240,54                          <064>
180 DATA 201,254,240,31,200,201,253,240,26
    ,200,201,251,240,21,200                    <081>
190 DATA 201,247,240,16,200,201,239,240,11
    ,200,201,223,240,6,200                     <072>
200 DATA 201,191,240,1,200,196,2,240,12,13
    2,2,185,225,3,164                          <240>
210 DATA 198,153,119,2,230,198,76,49,234,1
    69,255,133,2,208,247                       <064>
220 DATA 022:REM HELP                          <174>
230 DATA 056:REM 8                             <027>
240 DATA 053:REM 5                             <194>
250 DATA 023:REM TAB                           <070>
260 DATA 050:REM 2                             <117>
270 DATA 052:REM 4                             <193>
280 DATA 055:REM 7                             <046>
290 DATA 049:REM 1                             <163>
300 DATA 027:REM ESC                           <019>
310 DATA 043:REM +                             <240>
320 DATA 045:REM -                             <060>
330 DATA 017:REM LINE FEED                     <233>
340 DATA 013:REM ENTER                         <052>
350 DATA 054:REM 6                             <083>
360 DATA 057:REM 9                             <192>
370 DATA 051:REM 3                             <004>
380 DATA 024:REM ALT                           <053>
390 DATA 048:REM 0                             <230>
400 DATA 046:REM .                             <173>
410 DATA 145:REM CRSR UP                       <150>
420 DATA 017:REM CRSR DOWN                     <114>
430 DATA 157:REM CRSR LEFT                     <194>
440 DATA 029:REM CRSR RIGHT                    <069>
450 DATA 025:REM NO SCROLL                     <011>
```

© 64'er

Listing 30. Flexible Tastaturbelegung mit »ZUSATZTASTEN«. Eingabe erfolgt mit dem Checksummer 64.

Die ASCII-Codes der Tasten HELP, TAB, ALT und NO SCROLL habe ich ausgewählt, da diese Codes beim C 64 keine besonderen Aufgaben haben und somit gut in eigenen Programmen abgefragt werden können. Beispiel:
GET A$: IF A$ = CHR$(22) THEN...:REM HELP

Taste	ASCII-Code
0	48
1	49
2	50
3	51
4	52
5	53
6	54
7	55
8	56
9	57
.	46
+	43
—	45
HELP	22
TAB	23
ALT	24
ESC	27
LINE FEED	17
ENTER	13
CRSR UP	145
CRSR DOWN	17
CRSR LEFT	157
CRSR RIGHT	29
NO SCROLL	25

Tabelle 3. Zuordnung von ASCI-Codes zu den Zusatztasten

Die Tastenbelegung kann jedoch auch geändert werden. Dazu muß man nur die DATA-Werte in den Zeilen 220 bis 450 entsprechend ändern.

Das Programm belegt normalerweise den Kassettenpuffer, was jedoch bei Verwendung der Datasette ungünstig ist. Durch Ändern des Wertes in Zeile 10 kann das Programm auch an jeden anderen Platz »gelegt« werden. (Es belegt 189 Byte.)

Programmbeschreibung

Wollen wir verstehen, wie das Programm funktioniert, müssen wir uns auch dafür interessieren, wie normalerweise die Tasten abgefragt werden. Diese sind in Zeilen und Spalten aufgeteilt (Tabelle 4). Will man nun wissen, ob eine bestimmte Taste gedrückt ist, dann muß man durch Löschen des entsprechenden Bits in $DC00 (56320) die Zeile

		$DC01 (56321)							
	BIT	0	1	2	3	4	5	6	7
$DC00 (56320)	0	DEL	RET	CRSR→	F7	F1	F3	F5	CRSR↓
	1	3	W	A	4	Z	S	E	SHIFT links
	2	5	R	D	6	C	F	T	X
	3	7	Y	G	8	B	H	U	V
	4	9	I	J	0	M	K	O	N
	5	+	P	L	—	.	:	@	,
	6	£	★	;	HOME	SHIFT rechts	=	<	/
	7	1	←	CTRL	2	LEER-TASTE	C=	Q	RUN-STOP

Tabelle 4. Die Tastaturabfragematrix

auswählen, in der sich die Taste befindet. Für jede gedrückte Taste dieser Zeile wird nun das entsprechende Bit in $DC01 (56321) ebenfalls auf Null gesetzt. (Zu beachten ist, daß immer nur 1 Bit in $DC00 gelöscht sein darf, da man sonst nicht erkennen kann, in welcher Zeile sich die Taste befindet!)

Die Abfrage der »zusätzlichen« Tasten funktioniert im Grunde genauso. Diese sind nämlich auch in Zeilen und Spalten aufgeteilt (Tabelle 5). Die Zeilenauswahl wird hier jedoch durch die Bits 0 bis 2 von $D02F (53295) vorgenommen. Ist eine Taste gedrückt, wird wieder das entsprechende Bit in $DC01 gelöscht. Möchte man also eine der »zusätzlichen« Tasten abfragen, muß man darauf achten, daß alle Bits von $DC00 auf 1 gesetzt sind, da sonst das Drücken einer »normalen« Taste auch ein Bit in $DC01 löschen könnte! (Das gleiche gilt umgekehrt auch für die Abfrage der »normalen« Tasten!)

		$DC01 (56321)							
	BIT	0	1	2	3	4	5	6	7
$D02F (53295)	0	HELP	8	5	TAB	2	4	7	1
	1	ESC	+	—	LINE FEED	ENTER	6	9	3
	2	ALT	0	.	CRSR↑	CRSR↓	CRSR←	CRSR→	NO. SCR.

Tabelle 5. Die Tastaturmatrix der zusätzlichen Tasten

Das Programm ZUSATZTASTEN funktioniert nun auf folgende Weise: Es wird ein Maschinenprogramm generiert (Zeile 10 bis 50 und DATA-Zeilen), das in den Interrupt »eingebaut« wird (Zeile 60 bis 90). Dieses Programm fragt nun zirka 60mal in der Sekunde die »zusätzlichen« Tasten ab und schreibt deren ASCII-Code in den Tastaturpuffer. Die ASCII-Codes befinden sich in einer Tabelle am Ende des Programms und können, wie schon erwähnt, geändert werden. (J. H. Schulz/ef)

39. C128 und Final Cartridge

Kürzlich machte ich eine Entdeckung mit dem Modul Final Cartridge, die meiner Meinung nach auch andere C 128-Besitzer interessieren dürfte. Sie betrifft das Betreiben der Final Cartridge zusammen mit einem C 128.

Laut mitgelieferter Anleitung ist es nicht möglich, während das Modul im Expansion-Port steckt, in den C 128- oder den CP/M-Modus zu schalten. Doch nichts (fast nichts) ist unmöglich. Mit dem folgenden kleinen Trick schafft man auch dies:

1. Nach dem Einschalten des Computers befindet man sich im Desktop der Final Cartridge.
2. Mit dem Befehl Final Kill aus dem Desktop verläßt man dieses und kommt in den C-64-Modus.
3. Nun betätigt man nur noch die Reset-Taste des C 128 und man landet im C-128-Modus und kann dessen Software laden oder CP/M booten. Dieser Trick dürfte die relativ empfindliche Steckverbindung am Expansion-Port erheblich schonen. (Reto Meyer/ef)

40. Zehnerblock des C 128 für Btx nutzen

Mit einem kleinen Trick läßt sich der Zehnerblock des C 128 für den Betrieb mit dem Commodore Btx-Modul II nutzen. Ich habe dafür das Listing »10er-tast« aus dem Sonderheft 10 (das zweite Sonderheft für den C 128) benutzt.

Vorgangsweise:

Nach dem Abtippen des kleinen Hilfsprogramms 10ertast wird mit dem Maschinensprache-Monitor des C 128 ein jmp $1003 an das Programm angehängt. Das Ganze speichern Sie nun durch — S "BTX-EXTRA.MAS",8,c000,c08c — auf einer Diskette. Wenn man jetzt bei bereits eingeschalteter Floppy und angeschlossenem Btx-Modul II den Computer bei gedrückter CTRL-Taste einschaltet, so wird automatisch das Hilfsprogramm geladen und nach dem Loslassen der CTRL-Taste gestartet.

Die Eingabe einzelner Btx-Seiten oder die Auswahl innerhalb von Anbietermenüs gestaltet sich jetzt wesentlich einfacher und schneller. (Rainer Dahlhaus/ef)

41. SSHAPE korrigiert

Nach stundenlanger Fehlersuche in einem eigentlich korrekten Programm ist mir ein Fehler im Basic-ROM des C 128 aufgefallen, der, so meine ich, auch für andere C 128-Besitzer interessant sein kann. Dieser Fehler befindet sich in der Routine des SSHAPE-Befehls. Wenn man diesen Befehl verwendet, werden die Speicherstellen $1135 bis $ 1138 in die darunterliegenden Speicherstellen $1131 bis $1134 kopiert. Dies geschieht jedoch nicht, wie es richtig wäre, in der Speicherbank 0, sondern in der Speicherbank 1! Daß dies unangenehme Folgen auf eine an dieser Stelle gespeicherte Variable hat, dürfte wohl klar sein. Um das zu vermeiden, sollte man am Anfang eines Programms, in dem man den SSHAPE-Befehl verwendet, mit

```
POKE 47,53: POKE48,17: CLR
```

den Variablenanfang auf $1135 hochsetzen. Damit geht zwar der Bereich von $0400 bis $1130 für Variablen verloren, aber es wird sich sicherlich eine Verwendung für 3376 Byte Speicherplatz finden lassen. (Jan Hendrik Schulz/ef)

42. Protext leicht gemacht

Mein Tip bezieht sich auf das von Markt & Technik angebotene Textverarbeitungsprogramm »Protext«. Bei der Textausgabe durch <ESC> <S> wird bei gedrückter 40/80-Taste (also 80 Zeichen) der linke Teil des Textes auf dem Bildschirm angezeigt. Ist aber der 40-Zeichen-Modus eingeschaltet, so wird der rechte Teil des Textes, der sonst nicht sichtbar ist, auf dem Bildschirm angezeigt. Dieses steht nicht im Handbuch, ist aber sehr nützlich, wenn man den 120-Zeichen-pro-Zeile-Modus benutzt. Noch etwas Interessantes ist mir aufgefallen: Sollten Sie vor dem Laden vergessen haben die DIN-Tastatur einzustellen, müssen Sie Protext nicht neu laden. Die Funktionen, die sonst mit <Commodore> aufgerufen werden, können nun ohne jede Einschränkung über <Control> benutzt werden. Besonders für die 1541-Besitzer bedeutet dies eine deutliche Zeitersparnis. (C. Prenzel/ef)

43. Neuer Ton für <Control G>

Geben Sie im Direktmodus folgende Zeile ein:

```
SOUND1,10000,10000:VOL 0
```

Danach drücken Sie einmal die Tasten »Control+G«. Der Ton hat sich verändert. Sie können selbstverständlich auch jede andere Frequenz nehmen, um den Ton zu verändern. (H. Prang/ef)

44. SMON und der Blechdiesel

Beim SMON funktioniert auf dem neuen C 128D (Blech) der Trace-Modus nicht. Das System-Timing stimmt offensichtlich nicht, der Prozessor wird in seiner Rechenzeit vermutlich durch bestimmte Peripherie-Bausteine eingeschränkt. Vielleicht sind bei dem 8502-Mikroprozessor tatsächlich einige Befehle um einen Taktzyklus länger.

Zur Korrektur: An der Adresse $CD8D findet sich bei den verschiedenen SMON $C000-Versionen (bei anderer Startadresse an entsprechender Stelle) der Befehl »LDA #$47«. Wird dieser auf »LDA #$48« (ggf. #$49) geändert, so läuft der Trace-Modus wieder. Der geänderte SMON läuft natürlich auch auf einem normalen C 64.

Übrigens: Das Register 48 des VIC-Chip (Adresse $d030) dient nicht nur zum Umschalten zwischen 1 MHz und 2 MHz Taktfrequenz. Durch Setzen von Bit 1 wird der Bildschirm völlig abgeschaltet. Durch welche Adresse allerdings die Farbe nach dem Abschalten bestimmt wird, konnte ich noch nicht klären. Im Normalfall ist es grau. Wird das Bit im C 128-Modus gesetzt, so stürzt der Computer ab.

(T. Lang/ef)

45. NLQ-Schrift für Startexter 128 mit dem Citizen 120D

Im Sonderheft 22 wurde eine Erweiterung des Startexter 128 vorgestellt, die es erlaubt, den gesamten Zeichensatz im verbesserten Grafikdruck zu Papier zu bringen. Leider ist dieser Ausdruck mit dem Citizen 120D mit Original-Interface nicht möglich, da dieser Drucker das Steuerzeichen für »Backspace« ($08) als Commodore-Code für »bit Image« interpretiert. Durch Zufall konnte ich jedoch feststellen, daß »Backspace« durch ESC CHR$(8) zu erreichen ist (steht nicht im Handbuch).

Nun kann man die Datei »C0« auf einer Sicherungskopie folgendermaßen anpassen: File »C0« mit BLOAD laden und den Monitor aktivieren. Folgende Stellen ändern:

```
A 00DC9 LDA #$1B
A 00DCB JSR $0EDO
A 00EDO JSR $AB01
A 00ED3 LDA #$08
A 00ED5 JMP $AB01
```

Nun speichern Sie diese Datei nur noch mit folgenden zwei Befehlen auf Ihre Sicherungskopie:

```
@, S:CO
S"CO",8,COO,ED8
```

(H. Wiesler/ef)

46. Das zweite Betriebssystem im C 128

In den C 128 passen keine Umschaltplatinen für weitere Betriebssysteme des C 64-Modus. Deshalb habe ich eine einfache Lösung gefunden, mit der sich einfach (ohne Beschädigung des Abschirmblechs) zwei Betriebssysteme (zum Beispiel Original-Kernel und Exos) in ein 27256 brennen und in den Computer (nicht C 128 Blech!) einbauen lassen. Dabei geht man folgendermaßen vor: Beachten Sie, daß bei den C 128 der Basic-Interpreter und das Betriebssystem in einem Baustein vereint sind. Daher muß man den Interpreter mit in das EPROM brennen.

Man lädt also einen Monitor (SMON) nach $C000 und startet diesen mit SYS 49152. Nun verschiebt man den Interpreter mit W A000 C000 1000 und W A000 C000 5000. Der Interpreter wird für das normale und für das Betriebssystem gebraucht und deshalb zweimal verschoben. Danach wird das CBM-Kernel mit WE000 FFFF 3000 nach $3000 verschoben und die Adresse $4FFF mit dem Wert $FF geladen. Das ist nötig, weil der Smon das letzte Byte nicht mit verschiebt und den Befehl W E000 0000 3000 nicht akzeptiert. Also M 4FFF eingeben und das erste Byte in FF ändern. Nun wird das externe Betriebssystem nach $7000 geladen und zwar mit L"Filename"7000.

Nun steht das brennfertige Programm von $1000 bis $9000 im Speicher und kann mit S"Filename"1000 9000 auf Diskette gespeichert oder direkt in ein EPROM 27256 gebrannt werden. Ist das EPROM gebrannt, kann man schon mal den Lötkolben einschalten. Nehmen Sie das EPROM in die Hand und biegen Pin 27 (am besten mit einer Spitzzange) vorsichtig um 90 Grad nach oben. Dieser Pin darf später nicht mit in die Fassung gesteckt werden!

Nun verzinnt man Pin 1, Pin 14 und Pin 27 mittels Lötkolben und Lötzinn. Dies sollte möglichst schnell geschehen, um das EPROM nicht zu beschädigen (Lötkolben muß richtig heiß sein). Jetzt wird ein 10-KΩ-Widerstand (siehe Bild 1)

Bild 1. Die einfachste Betriebssystem-Umschaltung für den C64-Modus des C128

an Pin 27 und Pin 1 und der Schalter mittels zweipoliger Flachbandleitung an Pin 14 und Pin 27 gelötet. Ist diese Arbeit abgeschlossen, kann man das EPROM gegen das Original-Kernel austauschen (auf Kerbe achten) und zwischen den beiden Betriebssystemen mittels Schalter wählen.

Die Funktion der Schaltung ist einfach erklärt: Das Original-ROM ist bis auf Pin 27 Pin-kompatibel zu dem EPROM 27256. An Pin 27 des EPROM befindet sich Adreßleitung A14. Am Kernel-ROM jedoch ist dieses Pin ohne Funktion und liegt auf +5 Volt. Daher lassen sich am EPROM-Pin A14 die oberen oder die unteren 16 KByte auswählen. Der 10-KΩ-Widerstand dient dazu, daß Pin 27 bei offenem Schalter auf High-Pegel liegt. Bei geschlossenem Schalter liegt Pin 27 auf 0 Volt. (J. Woethe/ef)

47. Thirty-Seconds und der Blechdiesel

Damit auch die Besitzer eines C 128 D Blech das Backup-Programm Thirty-Seconds (Sonderheft 15, S. 96) nutzen können, sind einige Veränderungen vorzunehmen.

Da es sich nur um wenige Pokes handelt, können sie entweder im Direktmodus oder einfacher mit Hilfe eines kleinen Programms eingegeben werden:

```
10 key1,"poke 9739,106:poke 9843,142:poke 9919,130:
poke 9928,142:poke 10116,130:poke 11145,130:
poke 11249,142"+chr$(13)
```

Laden oder tippen Sie dieses Programm in den Rechner und starten Sie es mit RUN. Anschließend laden Sie thirty-seconds, drücken die Funktionstaste <F1> und speichern das Programm mit

```
dsave "thirty seconds 2"
```

(Frank Wurche/ef)

48. Ladeprogramm für Interlace-Funktionen

Besitzer eines C 128 D Blech können nun mit Hilfe weniger Änderungen das Interlace-Programm aus dem Sonderheft 29 nutzen. Dazu ersetzen Sie zunächst im Ladeprogramm in Zeile 180 den alten Wert (08192) für die Adresse durch 4096. Anschließend löschen Sie Zeile 540 und ergänzen das Programm um die in Listing 31 dargestellen Zeilen.

Speichern Sie nun den geänderten Lader unter dem Namen "INT.FKTN.LADER".

Im Hauptprogramm F(X).MAIN ist lediglich in Zeile 3410 RUN "INTERLACE.FKTN" durch RUN "INT.FKTN.LADER" zu ersetzen. Speichern Sie das Hauptprogramm auf derselben Diskette, auf der sich der neue Lader befindet.

(Clemens Thiele/ef)

```
540 POKE 246,255                                    <PK9>
550 PRINT CHR$(27)+"L": CHAR 1,30,20,"{GREEN}REVE
    RSED BY THE GREAT I.C.{54SPACE}": CHAR 1,24,1
    2: PRINT " ";: SLEEP 2: FAST                    <DBB>
560 FOR I=17 TO 75: WINDOW I,0,I,20: CHAR 1,0,20:
     FOR J=1 TO 20: : PRINT : NEXT J,I              <FP8>
570 SLEEP 2: WINDOW 0,0,79,24: SCNCLR : CLR : POK
    E 246,1: SLOW                                   <7CC>
580 DLOAD "F(X).MAIN"                               <M1K>
```

Listing 31. Diese Zeilen ergänzen das alte Ladeprogramm

49. HiRes-Hardcopy-Routine

Als Besitzer eines C 128 war ich schon lange auf der Suche nach einer geeigneten Hardcopy-Routine im HiRes-Modus. Mit großer Freude habe ich deshalb die Routine aus Sonderheft 29, Seite 131, in mein Zeichenprogramm, mit dem ich Schaltpläne erstelle, eingebaut. Leider war der Ausdruck nicht so gut, wie ich es erhofft hatte, da überflüssige senkrechte Striche das Bild unübersichtlich gestalteten. Zudem fehlte bei mir jeder achte Pixelpunkt im Ausdruck, was sich im Fehlen ganzer Linien auswirken konnte.

Nach längerem Überlegen habe ich nun eine Lösung (Listing 32) gefunden, die auch andere C 128-Besitzer bestimmt interessieren wird.

(Peter Burgbacher/ef)

```
100 OPEN 1,4: FAST                                  <OLM>
110 FOR T=316 TO 0 STEO -8                          <IM3>
120 SSHAPE B$,T,0,T+4,199                           <TVS>
130 SSHAPE C$,T+5,0,T+8,199                         <RTB>
140 DRAW 1,T,0 TO T,199                             <GNK>
150 SSHAPE A$,T,0,T+7,199                           <BSK>
160 R$=LEFT$(A$,200)                                <FQG>
170 PRINT#1,CHR$(8)+A$                              <0LJ>
180 GSHAPE B$,T,0: GSHAPE C$,T+5,0                  <JQM>
190 T=T+1                                           <99F>
200 NEXT                                            <180>
210 PRINT#1,CHR$(15)                                <URN>
220 SLOW : CLOSE 1                                  <FCL>
```

Listing 32. Die Hardcopy-Routine bringt alles gut zu Papier

Von Unidat Pro zu Mastertext

Mastertext und Unidat Pro gehören zu den Rennern unter der Software für den C 128. Mit diesem kurzen Listing ist es endlich möglich, Unidat-Dateien für Serienbriefe mit Mastertext zu nutzen.

Mastertext ist ein überaus leistungsfähiges Textverarbeitungsprogramm für den C 128. Im Sonderheft 29 stellten wir Ihnen die Version 2.0 vor, die sogar eine Rechtschreibprüfung (Speller) bietet. Im gleichen Heft veröffentlichten wir Unidat Pro, eine Dateiverwaltung mit riesigem Komfort.

Eine der zahlreichen Vorzüge von Mastertext ist das Erstellen von Serienbriefen. Der Anwender nutzt dabei bestehende Daten, zum Beispiel aus einer Adreßdatei, und fügt diese in einen Standardbrief ein. Der Vorteil dabei ist, daß ein Rundbrief nur einmal getippt wird, die zugehörigen Adressen (oder andere variable Daten) automatisch eingefügt werden.

Unidat Pro verfügt zwar über eine Funktion namens »Daten-Export«. Bei dieser werden gespeicherte Daten in eine sogenannte Exportdatei abgelegt. Es war bisher jedoch nicht möglich, Daten aus dieser Exportdatei für die Erstellung eines Serienbriefs zu nutzen. Das wird erst mit »Unidat Ex. Change« (Listing 1) erreicht.

Geben Sie das Listing 1 bitte mit dem Checksummer 128 ein und speichern es auf Diskette.

Nach dem Laden und Starten des Basic-Programms mit

```
RUN "UNIDAT EX.CHANGE"
```

fordert Sie das Programm auf, eine Diskette mit Unidat-Export-Dateien in das Laufwerk zu legen. Ist dies geschehen, drücken Sie bitte <SPACE>. Findet das Programm auf der eingelegten Diskette Dateien mit der Endung ».exp« (Kennung für die Unidat-Export-Dateien), werden diese rechts am Bildschirm aufgelistet. Auf der eingelegten Diskette sollten mindestens noch so viele Blöcke frei sein, daß die zu konvertierende ».exp«-Datei noch einmal gepeichert werden kann. Nach einem erneuten Drücken der SPACE-Taste erscheint die Frage »Export-Datei ?« und ein blinkender Cursor.

Geben Sie nun den Namen der Datei ein, die Sie in das Mastertextformat wandeln wollen. Bei der Eingabe ist die

Endung ».exp« nicht mit einzutippen. Anschließend ist der Name einzugeben, unter dem die gewandelte Datei auf Diskette gespeichert werden soll. Diese kann den Namen der Quelldatei tragen. Ist dies geschehen, beginnt die Wandlung; Sie erkennen dies daran, daß die Daten der Reihe nach in der linken Bildschirmhälfte erscheinen. Ist das Ende der Datei erreicht, springt das Programm zum Basic zurück, »READY.« und ein blinkender Cursor erscheinen auf dem Bildschirm.

Die konvertierte Datei ist nun auf der Diskette unter dem von Ihnen eingegebenen Namen, die Endung »R« fügt das Programm an diesen an. (Damian Plaza/rs)

Tips zu Unidat Pro

In der Anleitung zu Unidat steht, daß die Masken 255 Byte lang sein dürfen. Das ist nicht korrekt, die maximale Länge beträgt 254 Byte. Bei der Eingabe von mehr als 254 Datensätzen wunderten sich die Anwender, die die Version aus dem Sonderheft 29 besitzen, daß die Sätze über dieser Grenze nicht gefunden wurden. Mit folgender Änderung ist dies behoben:

1. Programm laden mit
DLOAD "UNIDAT PRO II.V"
2. Im Direktmodus eingeben:
POKE 11327,174:POKE 11330,240:POKE 11331,1:
POKE 11332,202:POKE 11333,236 <RETURN>
3. Ebenfalls im Direktmodus eingeben:
RENAME "UNIDAT PRO II.V" TO "UNIDAT ALT"
4. Und schließlich speichern:
DSAVE "UNIDAT PRO II.V"

Danach liegt die neue Version auf Diskette vor. Sollten Sie einen Fehler bei der Eingabe gemacht haben, besitzen Sie sicherheitshalber noch eine Sicherheitskopie unter den Bezeichnung <UNIDAT ALT>.

```
10 IF RWINDOW(2)=80 GOTO 30                              <PEN>
20 PRINT "{DOWN,RIGHT}AUF 80-ZEICHEN SCHIRM UMSCH
   ALTEN": SLEEP 1: PRINT CHR$(27)"X" CHR$(27)"M"        <IF6>
30 POKE 0,PEEK(0) OR 64: POKE 1,PEEK(1) AND 191:
   PRINT CHR$(14)                                        <E0F>
40 COLOR 6,7: COLOR 5,2: SCNCLR : FAST                   <2KT>
50 WINDOW 0,0,50,24,1: PRINT "{HOME,RVSON}DISKETT
   E MIT UNIDAT-EXPORT-DATEI(EN) EINLEGEN{RVOFF}"
   : GOSUB 260                                           <PPO>
60 WINDOW 51,0,79,21,1: COLOR 5,6: DIRECTORY "???
   ?????????.EXP": GOSUB 270: GOSUB 260: IF FE=1
   THEN RUN                                              <TQ9>
70 WINDOW 0,0,50,24,1: COLOR 5,2: INPUT "{DOWN,2R
   IGHT}EXPORT-DATEI ";EX$: EX=LEN(EX$): IF EX>12
    OR EX$="" THEN GOTO 70                               <NU4>
80 N1$=EX$+"{13SPACE}": N1$=MID$(N1$,1,12)+".EXP"
   : DOPEN #1,""+N1$+"": GOSUB 270: IF FE=1 THEN
   DCLOSE #1: GOTO 70                                    <DDC>
90 WINDOW 0,0,50,24,0: COLOR 5,2: CHAR 1,17,4,""+
   EX$+"": INPUT "{HOME,4DOWN,2RIGHT}MASTER-DATEI
    ";RU$: RU=LEN(RU$): IF RU>15 OR RU$="" THEN G
   OTO 90                                                <9EA>
100 N2$=RU$+"{16SPACE}": N2$=MID$(N2$,1,15)+"R":
    DOPEN #2,""+N2$+"",W: GOSUB 270: IF FE=1 THEN
     DCLOSE #2: DCLOSE #1: GOSUB 260: RUN                <F07>
110 WINDOW 0,12,79,24,1: COLOR 5,8                       <QF7>
120 GET #1,B$: C$="": IF B$="" THEN B$=CHR$(0)           <2SF>
130 IF ST=64 THEN GOTO 240                               <ECM>
140 PRINT B$;:                                           <A06>
150 B=ASC(B$)                                            <OQS>
160 IF B=187 THEN B=91                                   <96Q>
170 IF B=188 THEN B=92                                   <8IR>
180 IF B=189 THEN B=93                                   <9ES>
190 IF B=190 THEN B=94                                   <BQV>
200 IF B=219 THEN B=123                                  <KTU>
210 IF B=220 THEN B=124                                  <NHS>
220 IF B=221 THEN B=125                                  <N5V>
230 C$=CHR$(B): PRINT#2,C$;: GOTO 120                    <UGU>
240 PRINT#2,"@",CHR$(13): DCLOSE #2: DCLOSE #1           <2IH>
250 POKE 1,PEEK(1) OR 64: COLOR 6,1: COLOR 5,4: W
    INDOW 0,0,79,24: SCNCLR : PRINT CHR$(142): EN
    D                                                    <0NB>
260 A$="": DO WHILE A$="": GET A$: LOOP : RETURN         <DCC>
270 WINDOW 51,22,79,24,1: COLOR 5,1: FE=0: PRINT
    "{RVSON}"DS$"{RVOFF}": IF DS<>0 THEN PRINT CH
    R$(7): FE=1: RETURN : ELSE RETURN                    <HC5>
```

Listing 1. Mit »UNIDAT EX.CHANGE« wandeln Sie Unidat-Dateien in Mastertext-Rundschreiben-Dateien. Bitte mit dem Checksummer 128 (Seite 158) eingeben.

F U
FÜR

Den C 128-Fans steht ein riesiges Angebot an Literatur zur Verfügung. Hier eine Auswahl zu verschiedenen Themen, die Sie stets auf dem Weg vom Einsteiger zum Profi begleitet.

Bücher sind für den Computerfan die Butter auf dem Brot. Doch der Einkauf von Fachliteratur für Computer ist nicht so leicht, wie der im Supermarkt. Damit Sie sich nicht »den Magen verderben«, haben wir versucht, kurz und kritisch die Bücher für den C 128 darzustellen, die Ihnen den Weg vom Einsteiger zum Gourmet erleichtern. Guten Appetit!

Brot und Butter

Den Titel »C 128 für Einsteiger« kann man wörtlich nehmen. Der Leser wird Schritt für Schritt mit vielen Fotos und Bildern vom Auspacken und Aufstellen der Geräte bis zu ersten kleinen Basic-Programmen geleitet. Sehr gut für die Einsteiger geeignet, für die ein Computer noch ein grauer Kasten und ein Joystick etwas Unanständiges ist. »Vom C 64 zum C 128« (Bild 1) widmet sich besonders dem C-64-Piloten, der auf den C 128 umgestiegen ist. Angesprochen ist aber auch der C 128-Einsteiger, der schon erste Erfahrungen in Basic gemacht hat. Sehr informativ und gut verständlich beschreibt der Autor die Besonderheiten des C 128 und Unterschiede zum C 64 bezüglich Tastatur und Betriebssystem. Besonders ausführlich wird das Basic 7.0 des C 128 behandelt. Zahlreiche Tips und Tricks erfreuen das Herz des Programmierers. Assembler-Fans werden in einem eigenen Kapitel mit dem C 128-Monitor, dem »Tedmon«, vertraut gemacht. Auch gibt es ausführliche Hinweise zur Anpassung von C 64-Programmen auf den C 128. Ein Abschnitt über die Besonderheiten des C 64-Modus, ein Schnupperkurs zum Betriebssystem CP/M und ein kurzes Kapitel über die am C 128 verwendbaren Diskettenlaufwerke (die 1581 wird leider nicht erwähnt) runden das Bild ab. Ein Leckerbissen sind die vielen Beispielprogramme und Utilities auf der beiliegenden Diskette. Fazit: Besonders für Umsteiger ein sehr empfehlenswerter Zusatz zum Bedienungshandbuch des C 128. Einsteiger mit wenig Computererfahrung sind meiner Meinung nach mit einem der unten besprochenen Handbücher besser bedient. Das Original-C 128-Bedienungshandbuch ist – besonders für den Einsteiger – schwer verständlich und nicht ausführlich genug.



Bild 1. Das Aufsteiger-Handbuch

Drei Bücher wollen diesem Mißstand ein Ende bereiten (Tabelle 1): Der »Data Becker Führer« ist ein Nachschlagewerk. In tabellarischer Form führt es quer durch den C 128: Dazu gehören eine Auflistung und kurze Erläuterung sämtlicher Basic-Befehle, Sprungtabellen des Betriebssystems und des Basic; außerdem enthält das Buch eine Liste der CP/M-Befehle und -Besonderheiten, Registertabellen, Steckerbelegungen, Aufbau und Zeropage der Floppy 1571, die 6502-Prozessor-Befehle, Zeichencodetabelle des C 128, Fehlermeldungen und vieles mehr. Die Idee ist sicherlich sehr gut, doch leider gelang

TTER DEN GEIST

die Ausführung nicht ganz fehlerfrei.

So fehlen zum Beispiel bei den Basic-Befehlen Hinweise auf die fehlerträchtigen Anführungszeichen und auch eine Liste der nützlichen ESC-Kommandos ist nicht zu finden. Fazit: Symbolische Gedächtnisstütze für Freaks, die kein dickes Handbuch mehr mit sich herumschleppen wollen.

Über das »C 128-Buch« (Bild 2) sind nicht viele Worte zu verlieren. Es ist die deutsche Übersetzung des offiziellen Commodore-Programmierhandbuches aus den USA. Auf 670 engbeschriebenen Seiten wird der C 128 bis hin zum letzten Chip auseinandergenommen. Der Interessierte kann eine genaue Beschreibung der Basic 7.0-Befehle ebenso finden, wie Tabellen über den Stromverbrauch des C 128, oder der Gleich- und Wechselstromlasten der einzelnen Chips. Der Text ist sehr präzise und einprägsam. Für das Textverständnis sind Grundkenntnisse im Umgang mit Computern von Vorteil. Erwähnenswert ist das ausgezeichnete Stichwortverzeichnis am Schluß des Buches.

Fazit: Das Handbuch für den Spezialisten und Profi ist sein Geld wert.

»C 128 Intern« (Bild 3) wendet sich an den in Basic und Maschinensprache erfahrenen Leser. Es behandelt jeweils in eigenen Kapiteln den 40- und



Bild 2. Das Programmier-Handbuch



Bild 3. Ein Muß für Profis

den 80-Zeichen-Chip (VIC und VDC), die Ein- und Ausgabesteuerung, und – recht ausführlich – die Assemblerprogrammierung. Außerdem enthält es ein kommentiertes ROM-Listing des C 128-Kernel, eine Tabelle der Kernel-Routinen, eine Beschreibung der Speicheraufteilung und ein Kapitel über die Hardware mit symbolartigen Schaltplänen. In lockerem Stil versucht der Autor die Geheimnisse des C 128 aufzudecken. Allerdings sind für Einsteiger die Erläuterungen zu knapp und zu sehr gewürzt mit POKEs und Assemblerlistings. Fortgeschrittene dagegen müssen sich durch Textblöcke und Beispiellistings hindurcharbeiten, um die gewünschte Information zu erhalten. Ein Lob dem kommentierten Kernel-Listing, der kommentierten Zeropage und den Tabellen im Anhang.

Fazit: Wer sich zurechtfindet, erhält viele Tips und ein gutes Kernel-Listing.

Ein wenig Basic gefällig...?

»Das große Basic-Buch« von Data Becker ist im Moment das einzig verfügbare zu diesem Thema. Zuerst lernt der Leser, wie ein Problem überhaupt in ein Programm umzusetzen ist, um dann das Gelernte Schritt für Schritt in Basic zu verwirklichen. Am Ende jedes Kapitels befinden sich Übungsaufgaben, die helfen sollen, das Gelernte zu vertiefen. Es folgen kurze Kapitel über Dateiverwaltung, Grafik- und Musikprogrammierung, ein Einblick in die interne Basic-Verwaltung des Rechners und ein paar Tips und Utilities. Im Anhang gibt es eine sehr gute Übersicht der Basic-Befehle mit Erklärung der Syntax und nützliche Tabellen.

Fazit: Wer sich ernsthaft mit Basic beschäftigen will und keine Lust hat, sich nur mit einem Handbuch auseinanderzusetzen, ist mit dem »Großen Basic-Buch« bestens bedient.

Ein Sammelsurium nützlicher Programme, Pokes, Befehlserweiterungen, Assemblertricks, Einsprungadressen und Programmierhilfen bietet »Tips & Tricks für den C 128«



Bild 4. Für jeden etwas

(Bild 4). Es stur von vorn bis hinten durchzulesen, ist bei einem Buch dieser Art sicher nicht sinnvoll. Kaum jemand wird sich für sämtliche Tricks interessieren. Einzelne Tips mit Text und Inhaltsverzeichnis zu finden, erfordert jedoch Geduld. Wer mit einem guten Handbuch und einem ROM-Listing umzugehen weiß und manchmal das 64'er-Magazin aufschlägt, braucht sich eigentlich nicht mit diesen »Tips und Tricks« auseinanderzusetzen. Wem Zeit und Lust fehlt, sich umständlich durch Handbuch und ROM-Listing zu wühlen, sollte vor einem Kauf genau untersuchen, ob »Tips und Tricks« die eigenen Ansprüche erfüllt.

»Peeks und Pokes« will Basic-Programmierern helfen, die Tiefen des C 128 auszuloten. In flüssigem Text werden dem Leser die Speicherverwaltung und die Grafik- und Soundchips genau erklärt. Außerdem enthält das Buch eine kurze, gute Einführung in die Maschinensprache mit Beispielen am 8502- und Z80-Prozessor(!). Im Anhang finden sich die Assemblerbefehle beider Prozessoren, eine Speicherübersicht des C 128 und Registertabellen der Grafikchips VIC und VDC. Erarbeitete Programmbeispiele oder Tricks werden am Ende der jeweiligen Textblöcke noch einmal zusammengefaßt.

Fazit: Genau das Richtige für den Basic-Programmierer, der sich mit dem C 128 gut auskennt und nicht weiß, ob er in Maschinensprache einsteigen oder weiter in Basic programmieren soll (Tabelle 2).

... oder soll es etwas Assembler sein?

Für diejenigen, die ihrem C 128 mit Assembler auf den Leib rücken wollen, bieten sich zwei Einführungskurse an (Tabelle 3).

Beide enthalten je einen komfortablen Assembler auf Diskette, der keine Wünsche offen läßt, und bieten eine gute, ausführliche Einführung in Maschinensprache und Assembler-Programmierung. Sie unterscheiden sich nur durch die Bedienung des Assemblers und den Textaufbau.

Im »C 128 Assembler-Kurs« (Bild 5) arbeitet man von Anfang bis zum Ende mit dem mitgelieferten Assembler. Das hat den Vorteil, daß man sich am Ende sehr gut mit seiner Bedienung auskennt. Der Text ist präzise und ohne Schnörkel. Aufgaben am Ende jedes Kapitels vertiefen das Gelernte. Die Assemblerbedienung ist etwas gewöhnungsbedürftig. Sie erfolgt mittels eigener Befehle von Basic aus. Das Assembler-Programm wird ohne Zeilennummern eingegeben (Full-Screen-Editor), ist aber sehr



Bild 5. Assembler mitgeliefert

übersichtlich dargestellt. Die auf Diskette mitgelieferten fertigen Makros (Assembler-Unterprogramme, die bequem mit ihrem Namen aufgerufen werden) erleichtern das Erstellen eigener Programme.

»Programmieren in Assembler mit Top-Ass plus« (Bild 6) ist in zwei Teile gegliedert. Im ersten Teil, dem eigentlichen Lehrbuch, lernt man die Befehle und Möglichkeiten der Maschinensprache mit dem C 128-Monitor kennen.

Erst im zweiten Teil wird man mit der Bedienung des Assemblers vertraut gemacht. Hier liegt der Vorteil darin, daß der Einsteiger seine Beispielprogramme nachprüfen kann, ohne sie erst vorher zu assemblieren. Als Eingeweihter kann er später mit einer Art Handbuch arbeiten, ohne ausführliche Kursteile immer wieder durchgehen zu müssen. Der locker geschriebene Text ist leicht verständlich. Der Assembler arbeitet mit einem Zeileneditor.

Fazit: Probelesen! Mit beiden Büchern läßt sich sehr gut Maschinensprache lernen. Es kommt nur darauf an, welcher Textaufbau und welche Assemblerbedienung Ihnen am liebsten ist.



Bild 6. Kompletter Programmierkurs

CP/M öffnet unserem C 128 das Tor zu professioneller Software wie Multiplan und Wordstar oder dem legendären Turbo-Pascal, und bietet damit die faszinierende Möglichkeit, Daten auch mit anderen Computersystemen auszutauschen.

CP/M – Das etwas andere Betriebssystem

Für den ersten Kontakt mit CP/M bieten sich dem C 128-Besitzer drei Bücher an (Tabelle 4): Alle drei beschäftigen sich eingehend mit der Geschichte von CP/M, der Struktur von Befehlen und Dateien und der Bedienung der Peripherie.

Das »CP/M 3.0 Anwender Handbuch« (Bild 7) kommt einem Lehrbuch am nächsten.



Bild 7. Übersicht für Anwender

Mit flüssigem, großgedrucktem Text wird der Leser in die Grundlagen von CP/M eingeweiht. Viele Beispielprogramme und eine gute Befehlsübersicht im Anhang sind dabei von großer Hilfe.

»C 128 Arbeiten mit CP/M Plus« (Bild 8) beschreibt die Struktur und Bedienung von CP/M etwas nüchterner. Sehr gut gefallen hat mir die Liste der einzelnen Befehle mit ihren knappen, aber treffenden Erklärungen. Zusätzlich gibt es kurze Einführungen zu NEVADA, NEVADA-Fortran, Microsoft-Basic, Microsoft-Fortran-80 und Turbo-Pascal. Ein gutes Handbuch und Nachschlagewerk.

»Alles über CP/M 3.0« (Bild 9) bietet eine Vereinigung von Lehrbuch und Handbuch. In präzise formulierter Sprache



Bild 8. Das Nachschlagewerk

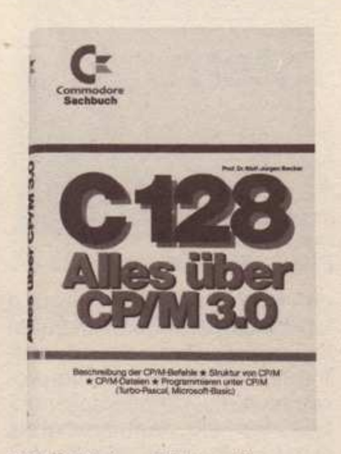


Bild 9. Intensivkurs für Einsteiger

wird der Leser mit Informationen überhäuft. Wer sich hier durchgearbeitet hat, weiß einiges mehr über CP/M. Eine sehr gute Befehlsübersicht, ein Anhang mit den wichtigsten Tabellen und ein gutes Inhalts- und Stichwortverzeichnis rüsten das Lehrbuch zum Handbuch auf.

Fazit: Wer hauptsächlich im C 128-Modus arbeitet und sich auch mit CP/M beschäftigen will, sollte sich, je nach Geschmack, Lehrbuch oder Handbuch anschaffen. Für all diejenigen, die CP/M intensiv nutzen wollen, ist »Alles über CP/M« sicher kein Fehlkauf. (Tabelle 4)

Grafik – Nur für Feinschmecker?

Der Titel »Grafikprogrammierung auf dem C 128« (Bild 10) sagt eigentlich schon alles. Die Grafikbefehle des Basic 7.0 fehlen in diesem Buch ebensowenig wie ausführliche Kapitel über Sprites und Shapes. Sehr schön sind die Kapitel über Fraktale und über die Anwendungen des Tastaturpuffers. Viel Platz wird den beiden Videochips gewidmet. Leider wird die hochauflösende Grafik im 80-Zeichen-Modus etwas stiefmütterlich behandelt. Es fehlen Hinweise zur Programmierung von Interlace-Bildern und zum 64-KByte-Speicher des VDC-Chips im neuen C 128 D (Blech). Dafür



Bild 10. Faszination der C 128-Grafik

gibt es sämtliche Beispielprogramme auf der beigelegten Diskette und als Schmankerl ein Programm für die Nutzung der Basic-7.0-Grafikbefehle auch auf dem 80-Zeichen-Bildschirm. Der flüssige, ansprechende Text erhöht den Lesespaß und macht das Buch sehr empfehlenswert für Einsteiger mit Grafikambitionen. Für Fortgeschrittene mit gutem Handbuch gibt es wenig Neues.

Pascal ist eine der besten Hochsprachen, um strukturiertes Programmieren zu erlernen, und die Zahl ihrer Fans steigt unaufhaltsam.

»Pascal mit dem C 128« (Bild 11) führt den Leser von den Grundbegriffen der Sprache hin zu komplizierten Anwendungen von Pascal in Kombination mit Maschinensprache. Ein Pascal-Compiler mit komfortablem Editor und Fehlerde-

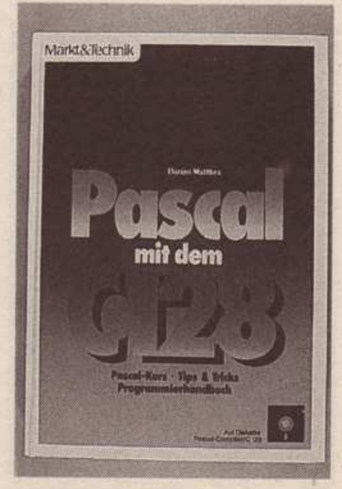


Bild 11. Einführung mit Compiler



Bild 12. Alles über die 1570/71

tektor liegt dem Buch zusammen mit vielen Beispielprogrammen auf Diskette bei. Der übersichtliche Text steigt schnell und intensiv in die Materie ein und erfordert Grundkenntnisse im Umgang mit Computern und in Mathematik. Die vielen Aufgaben am Ende der Kapitel sind wichtig für das weitere Textverständnis und sollten unbedingt bearbeitet werden. Eine ausführliche Dokumentation des angewandten Pascal-Systems 2.0 macht das Buch auch für den fortgeschrittenen Pascal-Anwender wertvoll. Eine Anwendung des schnelleren Turbo-Pascal-Compilers sollte mit den hier erworbenen Grundlagen keine wesentlichen Schwierigkeiten mehr bereiten. Fazit: Eine gute Alternative zu den meist trockenen und nicht auf den C 128 bezogenen Lehrbüchern zu Turbo-Pascal unter CP/M. Für Einsteiger nur zu empfehlen, wenn genügend Durchhaltevermögen vorhanden ist.

Das Diskettenlaufwerk C 1571 ist wegen seiner hohen Speicherkapazität und der Fähigkeit, unterschiedliche Diskettenformate zu bearbeiten, das am meisten verwendete Peripheriegerät an unserem C 128. Deshalb hier zwei Bücher über dieses Laufwerk (Tabelle 5). Leider gibt es noch keine Literatur über das äußerst leistungsfähige 3½-Zoll-Laufwerk 1581!.

»Das große Floppybuch« (Bild 13) enthält alles wesentliche, das man über seine Floppy wissen sollte. Der flüssig geschriebene Text behandelt das Auspacken des Gerätes ebenso, wie die Benutzerbefehle, die Floppy unter CP/M, die Besonderheiten der verschiedenen lesbaren Diskettenformate und die Fehler des DOS 3.0. Ein gut kommentiertes ROM-Listing des DOS schließt sich an. Die neue Auflage enthält auch endlich ein Stichwortverzeichnis und den Assembler-Quelltext, beziehungsweise den Data-Lader des recht leistungsfähigen Diskettenmonitors, der auch auf Diskette für DM 29.-(!) zusätzlich zu haben ist.

Das Buch »Die Floppy 1570/71« (Bild 12) ist vom Inhalt her mit dem »Großen Floppybuch« vergleichbar. Der ausführliche, gut lesbare Text führt quer durch sämtliche Möglichkeiten des Laufwerks. Sehr gut gefiel das ausführliche Kapitel über die User-0-Befehle, die zusätzliche Auflistung der Fehler von DOS und Handbuch in einem eigenen Kapitel und die detaillierte Beschreibung der RAM-Aufteilung im Anhang. Auch ein gut kommentiertes ROM-Listing und ein leistungsfähiger Diskettenmonitor zum Abtippen fehlen nicht. Der Diskettenmonitor ist hauptsächlich in



Bild 13. Floppy-Handbuch für Profis

Basic geschrieben, so daß später relativ einfach eigene Verbesserungen vorgenommen werden können. Durch die häufige Verwendung der schnellen Burst-Befehle wurde trotz Basic eine passable Geschwindigkeit erreicht. Keines der beiden Bücher beschäftigt sich mit den Unterschieden der neuen 1571 im »Blechdiesel« zum alten Modell. Interessierten sei das 64'er-Sonderheft-29 empfohlen. Fazit: Wieder muß der eigene Geschmack entscheiden. Beide Bücher informieren umfassend über die Floppy 1570/71. Wegen dem größeren und übersichtlicheren Format und den ausführlicheren Erklärungen im Text würde ich Einsteigern eher »Die Floppy 1570/71« empfehlen. Aber urteilen Sie selbst. (Tabelle 5)

Wer die Wahl hat...!

Es gäbe noch viele Bücher, die zu besprechen lohnen würde, wie zum Beispiel Handbücher zu Protext und dBase II oder Spezialitäten zu CP/M. Wer sich dafür interessiert, sollte sich genügend Zeit nehmen und im Buchhandel ausgiebigst Vergleiche ziehen. Vieles überschneidet sich im Inhalt oder hat einen für den eigenen Geschmack unpraktischen Textaufbau. Diese Übersicht soll Ihnen helfen, eine grobe Vor-Auswahl zu treffen.

(Alric Rüther/so)

Titel	Autor	DM	Disk	Verlag
1. Commodore 128 für Einsteiger	Szczepanowski	29.-	—	Data Becker
2. Vom C 64 zum C 128 Tips & Tricks	F.Müller	49.-	+	M. & T.
3. Der Data Becker-Führer zum C 128	H.Wrobel	19.80	—	Data Becker
4. Das C 128-Buch	L.Greenly u.a.	75.-	—	Sybex
5. Commodore 128 Intern	Schieb u.a.	69.-	—	Data Becker

Tabelle 1. Diese Bücher eignen sich, um einen schnellen Einstieg zum C 128 zu finden

Titel	Autor	DM	Disk	Verlag
6. Das große Basic-Buch zum C 128	Kampow	39.-	—	Data Becker
7. Commodore 128 Tips und Tricks	Hornig u.a.	49.-	—	Data Becker
8. Peeks & Pokes zum C 128	Liesert u.a.	29.-	—	Data Becker

Tabelle 2. Für den Basic-Programmierer gibt es drei Bücher

Titel	Autor	DM	Disk	Verlag
9. C 128 Assembler-Kurs	F.U.Müller	75.-	+	Sybex
10. Programmieren in Assembler mit TOP-ASS plus für den C 128	G.Möllmann u.a.	59.-	+	M. & T.

Tabelle 3. Zwei Kurse gibt es für den Assembler-Freak

Titel	Autor	DM	Disk	Verlag
11. CP/M 3.0 Anwender Handbuch C 128	J.Hückstädt	52.-	—	M. & T.
12. C 128 Arbeiten mit CP/M Plus	B.Bachmann	48.-	—	Sybex
13. C 128 Alles über CP/M 3.0	Prof.Dr.Becker	52.-	—	M. & T.

Tabelle 4. CP/M mit dem C 128 öffnet das Tor zu professioneller Software

Titel	Autor	DM	Disk	Verlag
14. Grafikprogrammierung C 128	H.Ponnath	52.-	+	M. & T.
15. Pascal mit dem C 128	F.Matthes	52.-	+	M. & T.
16. C 1570/71 Das große Floppybuch	Ellinger	49.-	—	Data Becker
17. Die Floppy 1570/1571	K.Schramm	52.-	—	M. & T.

Tabelle 5. Daneben sind Bücher zu den unterschiedlichsten Themenbereichen erhältlich

SUPER



Super-Vectors ist ein spannendes Action-Spiel für den C 128, das den Joystick heißlaufen läßt und Sie sicher für viele Stunden an den Bildschirm fesselt. Mit der Auflösung von 736 x 354 Bildpunkten bricht es alle Grafikrekorde.

DURCH´S NADELÖHR

CTORS

ZUM SIEG

Dieses sehr schnelle Geschicklichkeitsspiel fasziniert jeden, der es einmal gesehen hat. Es greift die Idee zweier sich bekämpfender Schlangen auf (ähnlich der Motorrad-Sequenz im Film »Tron«) und setzt sie auf dem C 128 im 80-Zeichen-Modus auf faszinierende Weise neu um. Bei »Super-Vectors« kommt es darauf an, den Gegner durch geschicktes Manövrieren einzukreisen, ihn gegen eine Wand rennen zu lassen und damit auszuschalten. Durch die hohe Grafikauflösung (736 x 354 Punkte) ist der Kampf der beiden Linien wesentlich spannender als bei bisherigen Versionen. »Super-Vectors« ist nur auf einem C 128 mit erweitertem VDC-RAM lauffähig, da die gesamten 64 KByte ausgenutzt werden.

Diese Voraussetzung erfüllt jeder C 128 im Blechgehäuse, aber auch andere C 128 sind umrüstbar auf den vollen Speicher.

Super-Vectors wurde mit einigen Zusatzfunktionen ausgestattet. So kann man als weitere Schwierigkeiten zufällige Hindernisse in das Spielfeld einschalten, die auch vom Benutzer editierbar sind. Auch diagonale Linien sind möglich, wodurch die Schwierigkeit für Vector-Profis gesteigert werden kann. Die Farben des Hintergrundes und der Zeichenfarbe sind zu vertauschen.

Es besteht auch die Möglichkeit, gegen den Computer anzutreten und verschiedene Schwierigkeiten anzuwählen. Diese können durch Ab- und Hinzuschalten verschiedener Computerstrategien zusätzlich variiert werden.

Hat man den Menüpunkt »Beschleunigen« gewählt, so läßt sich durch Drücken des Feuerknopfes der Vector (Linie) beschleunigen.

Super-Vectors bietet aber noch weitere Extras:

Zeitmessung:

Bei jedem Durchgang wird die Zeit bis zum »Crash« gestoppt. Die abschließende Bewertung stellt neben den Siegen auch die benötigte Durchschnittszeit gegenüber. Bei gleicher Anzahl der Siege beider Spieler entscheidet dieser Durchschnittswert über den Gesamtsieg.

SPACE PRESS

Aus technischen Gründen ist die Höchstzeit eines Durchgangs auf vier Minuten begrenzt. Haben Sie erst einmal einige Runden gespielt, werden Sie schnell feststellen, daß dieser Grenzwert so gut wie nie erreicht wird.

Zufallsstart:
Bei Anwahl dieser Option starten die beiden Vektors an beliebigen Positionen und mit zufälliger Richtung. Dies stellt eine besondere Herausforderung an Reaktion und Spielvermögen dar, zumal zuerst herauszufinden ist, welchen der beiden Vektors man überhaupt steuert.

Als geradezu heimtückisch erweist sich die Kombination des Zufallsstarts mit der Option »Hindernisse ein«. Ein Vektor kann kurz vor einer Begrenzungslinie starten oder von einem Hindernis gefangen sein. Um ein schnelles Ende nicht zur Regel werden zu lassen, wird den Spielern anfangs eine halbe Sekunde zur eigenen Orientierung eingeräumt.

Geschwindigkeit:
20 verschiedene Stufen sind einstellbar, die gerade bei höherem Tempo feine Unterschiede erlauben.

Abbruch:
Die Restore-Taste unterbricht einen Durchgang. Sie gelangen mit <RESTORE> zum Hauptmenü.

Ende:
Man kann das Programm verlassen, ohne den Computer ausschalten zu müssen. Dies ist vor allem dann wichtig, wenn Sie selbstkonstruierte Hindernisse mit einem Maschinensprache-Monitor modifizieren wollen (siehe unten).

Computerstrategien

Die Spielstärke des Computers wird durch Hinzuschalten verschiedener Strategien eingestellt. Im Programm sind die zehn sinnvollsten Kombinationen der folgenden Einzelstrategien integriert:
- Spirale
- Gasse
- Folgen
- Besoffen
- Zufall

Der Computer-Vector ist prinzipiell so gesteuert, daß er zwei Pixel vor einem Hindernis (Gegner-Vector, Hindernis oder Spielfeldrand) ausweicht. Durch dieses Freilassen eines Pixels sichert er sich einen Ausweg, falls er in eine Falle läuft.

Beim Ausweichen wird die neue Richtung zufällig ausgewählt (siehe auch »Folgen«).

Die Strategien im einzelnen

Spirale: Diese Strategie überprüft, ob sich der Computer-Vector in einer Spirale verfangen hat und sichert sein Entkommen.

Gasse: Steuert der Computer auf einen Spalt zu, der nur einen Punkt breit ist (mögliche Sackgasse), so weicht er aus.

Folgen: Bei einem Ausweichmanöver orientiert sich der Computer an der aktuellen Position des Gegner-Vectors. Man wird von ihm verfolgt!

Besoffen: Der Computer weicht nach einer bestimmten Anzahl zurückgelegter Pixel aus (siehe auch »Folgen«), egal, ob er auf ein Hindernis stößt oder nicht. Damit wird er unberechenbar!

Zufall: Diese Strategie arbeitet mit »Besoffen« zusammen. Sie bestimmt die Anzahl der Pixel, nach der der Computer die Richtung wechselt. Falls die Strategie aktiviert ist, addiert er zu einem Startwert einen Zufallswert, ansonsten übernimmt er den Startwert.

Die Spielstärke ergibt sich nicht unmittelbar aus der Reihenfolge der Strategiekombinationen. Unserer Meinung nach ist die Strategie 7 die spielstärkste. Sehenswert ist aber auch Strategie 4.

Ändern der Hindernisse

Nach dem Laden des Programms befindet sich im Speicher eine Tabelle. Diese Tabelle liegt ab der Speicherstelle $03026 in BANK 0.

In der Tabelle 2 sind für jedes Hindernis zwei Byte reserviert, die den Anfang eines Hindernisses in Low-Byte und High-Byte angeben:

Byte 0 und 1: Low-Byte und High-Byte der Anfangsadresse des Hindernisses Nummer 0;

Byte 2 und 3: Low-Byte und High-Byte der Anfangsadresse des Hindernisses Nummer 1;

und so weiter...

Das Programm erkennt ein ausgeschaltetes Hindernis daran, daß die zugehörige Adresse den Wert Null besitzt.

Die maximal sechs zusätzlichen Hindernisse können ab der Adresse $033a2 in BANK 0 abgelegt werden.

Aufbau eines Hindernisses

Ein Hindernis baut sich aus vertikalen, horizontalen und diagonalen Strichen auf. Ein Strich setzt sich aus sechs Byte zusammen:
- Byte 0: Low-Byte der X-Koordinate der Startkoordinaten des Striches
- Byte 1: High-Byte der X-Koordinate der Startkoordinaten des Striches
- Byte 2: Low-Byte der Y-Koordinate der Startkoordinaten des Striches
- Byte 3: High-Byte der Y-Koordinate der Startkoordinaten des Striches
- Byte 4: Länge des Striches (maximal 255)
- Byte 5: Richtung des Striches
  - 0 – Rechts
  - 1 – Unten
  - 2 – Links/Unten
  - 3 – Rechts/Unten

Ein Strich mit den Koordinaten X=317, Y=322 und der Länge 46 in Richtung Rechts/Unten sähe so aus:

Byte:	0	1	2	3	4	5
Wert:	061	001	097	066	001	003

Zum Austesten empfiehlt es sich, den eingebauten Monitor zu benutzen.

Eingabe des Programms

Das Hauptprogramm (Listing 1) geben Sie mit dem MSE in C64-Modus ein und speichern es unter dem Namen »Supervectors« auf Ihren Datenträger. Danach drücken Sie bitte den Reset-Knopf oder schalten den Computer aus und ein, um in den C128-Modus zu gelangen.

Das Programm belegt den Bereich ab dem Basic-Anfang ab $1c01. Es ist mit »RUN "Supervectors"« zu laden und startet dann automatisch.

Zu Beginn ist der Monitor so einzustellen, daß der Rand des Spielfelds rundum sichtbar ist. Sollte das Bild »durchlaufen«, stoppen Sie es mit dem Regler mit der Bezeichnung »V.hold«. Dieser Regler ist allerdings an einigen Monitoren nicht mehr außen angebracht.

Die Bewegung der Vectors und die Auswahl der Menüpunkte erfolgt mit dem Joystick (80-Zeichenschirm einschalten), erster Spieler in Port 2.

Wem die Hindernisse und Strategien trotzdem noch zu leicht sind, der sollte ruhig das Spiel erweitern. So zum Beispiel einen dritten, unabhängigen Vector, der jeden der beiden anderen behindert. Oder knobeln Sie selbst noch weitere, ausgekochte Hindernisse aus.

(Frank Heidermanns/rs)

Super-Vectors basiert auf dem Programm »Vectors«, das die Autoren H. Beine und A. Janssen geschrieben haben. Es wurde im Sonderheft 22 veröffentlicht.

```
Name : super-vectors      1c01 33a2
-----------------------------------
1c01 : 0b 1c c4 07 9e 37 31 38  05
1c09 : 34 00 00 00 20 4b 24 a9  7e
1c11 : 00 8d 00 ff 20 b6 77 a2  b3
1c19 : ff 9a e8 8e 02 dc 8e 03  b8
1c21 : dc e8 86 28 86 68 20 78  35
1c29 : 1e 20 8b 2c a2 19 20 da  e9
1c31 : cd 29 07 09 80 20 cc cd  4e
1c39 : a0 00 20 7c 2c 4c 9e 25  5b
1c41 : 20 03 c4 ad 20 1b c9 ef  ab
1c49 : f0 01 60 ad 18 1b 8d 23  5e
1c51 : 1b a9 00 85 41 85 42 85  46
1c59 : 43 85 44 8d 24 1b 8d 25  bd
1c61 : 1b a2 0c 20 cc cd 20 7e  8d
1c69 : 1e 20 71 2c a9 32 85 a9  0f
1c71 : a2 01 86 6b 86 6c 86 aa  de
1c79 : ca 86 ab 8e 31 1b 8e 32  ce
1c81 : 1b 8e 00 1b 8e 01 1b 8e  c1
1c89 : 02 1b a0 1c 20 7c 2c 20  9b
1c91 : 34 21 20 77 2c a9 c0 a2  a5
1c99 : 2c 20 40 2a a5 6a 29 20  bd
1ca1 : f0 11 ad 06 dc 29 1e a8  27
1ca9 : be 27 30 f0 f5 b9 26 30  4b
1cb1 : 20 40 2a 24 af 10 69 a2  67
1cb9 : 00 86 24 86 26 ad 12 d0  90
1cc1 : ac 04 dc 20 0b 1d 20 bb  3c
1cc9 : 1d 20 c2 1e b0 e9 20 c4  cf
1cd1 : 1e a2 00 86 64 86 66 a2  6b
1cd9 : 40 ad 06 dc ac 12 d0 20  ec
1ce1 : 0b 1d 20 c6 1d c5 23 f0  ca
1ce9 : e8 c4 25 f0 e4 20 c2 1e  31
1cf1 : b0 df 20 c4 1e a2 01 bd  a8
1cf9 : 06 dc 29 03 a8 b9 3c 2a  b6
1d01 : 95 8e ca 10 f2 20 03 c4  58
1d09 : f0 3a 29 aa 0a 36 24 0a  ad
1d11 : 36 24 d0 02 09 40 95 23  fd
1d19 : 98 69 32 95 25 36 26 60  02
1d21 : a9 32 85 23 a9 00 85 24  a2
1d29 : a9 ae 85 63 a9 02 85 64  81
1d31 : a9 b1 a2 00 85 25 86 26  43
1d39 : 85 65 86 66 a9 f7 85 8e  6d
1d41 : a9 fb 85 8f 8e 0a dd 8e  09
1d49 : 09 dd 8e 08 dd e8 86 ad  80
1d51 : 86 a8 20 bb 1d 20 c4 1e  cd
1d59 : a2 00 20 fb 20 20 d1 1d  07
1d61 : 90 0b 20 bb 1d 20 c2 1e  10
1d69 : 90 03 4c 07 22 a5 6a 29  ba
1d71 : 10 f0 08 20 41 1f 8d 32  a7
1d79 : 1b d0 05 a2 01 20 fb 20  d3
1d81 : a2 01 20 d1 1d 20 c6 1d  0e
1d89 : 90 08 20 c2 1e 90 03 4c  89
1d91 : 0a 22 20 c4 1e ad 09 dd  7c
1d99 : c5 ad f0 0f 85 ad 20 45  24
1da1 : 1e ae 0a dd e0 04 90 03  cb
1da9 : 4c 0f 22 a6 27 bc 27 2a  24
1db1 : a2 01 ca d0 fd 88 d0 fa  fe
1db9 : f0 98 a5 26 85 1b a5 23  32
1dc1 : a6 24 a4 25 60 a5 66 85  1f
1dc9 : 1b a5 63 a6 64 a4 65 60  26
1dd1 : 24 6a 10 0e a8 29 10 f0  e6
1dd9 : 08 d6 6b d0 65 a9 02 95  18
1de1 : 6b 98 24 6a 50 02 09 10  48
1de9 : a8 18 75 8e c9 fb f0 12  31
1df1 : c9 f2 f0 0e 98 a0 03 d9  7f
1df9 : 3c 2a f0 04 88 10 f8 2c  4c
1e01 : 95 8e b4 8e 8a f0 02 a2  5a
1e09 : 40 c0 fe d0 0d d6 28 30  0c
1e11 : 2d b5 25 d0 02 d6 26 d6  99
1e19 : 25 60 c0 fd d0 0b d6 28  6f
1e21 : 30 1c f6 25 d0 02 f6 26  07
1e29 : 60 c0 fb d0 09 b5 23 d0  6f
1e31 : 02 d6 24 d6 23 60 f6 23  da
1e39 : d0 02 f6 24 38 60 a9 02  7e
1e41 : 95 28 18 60 a2 01 8e c0  ea
1e49 : 1b ca 8e c7 1b a9 2c 8d  31
1e51 : c1 1b 8e c2 1b ad 0a dd  9f
1e59 : 48 a9 0a 48 ad 09 dd aa  f1
1e61 : 4a 4a 4a 4a 48 8a 48 a0  e7
1e69 : 03 68 29 0f 09 30 99 c3  cd
1e71 : 1b 88 10 f5 4c 14 2c a9  fd
1e79 : 40 a2 fa d0 04 a9 0d a2  ea
1e81 : 1c 8d 18 03 8e 19 03 60  49
1e89 : a5 ac d0 0b a5 a6 d0 02  f1
1e91 : c6 a7 c6 a6 4c b8 1e c9  48
1e99 : 01 d0 08 e6 a6 d0 02 e6  a8
1ea1 : a7 b0 14 c9 02 d0 0a a5  f9
1ea9 : a4 d0 02 c6 a5 c6 a4 b0  93
1eb1 : 06 e6 a4 d0 02 e6 a5 a5  a7
1eb9 : a7 85 1b a5 a4 a6 a5 a4  fe
1ec1 : a6 18 24 38 85 1e 86 1f  25
1ec9 : 84 20 48 29 07 85 22 8a  cf
1ed1 : 48 98 48 08 a2 03 46 1f  12
1ed9 : 66 1e ca d0 f9 a5 1b a2  99
1ee1 : 05 06 20 2a ca d0 fa a6  a3
1ee9 : 20 86 1c 85 1d 06 20 2a  db
1ef1 : 85 1f a5 1c 65 20 85 20  a0
1ef9 : a2 0c 20 da cd 65 1d 65  4c
1f01 : 1f 85 21 a5 1e 20 37 1f  de
1f09 : 20 2a 1f 20 d8 cd a6 22  e5
1f11 : 28 90 0d 1d 6c ce 48 20  07
1f19 : 2a 1f 68 20 ca cd b0 06  db
1f21 : 3d 6c ce f0 01 38 4c 3a  dd
1f29 : fa a2 12 a5 21 20 cc cd  8f
1f31 : e8 a5 20 4c cc cd 18 65  e4
1f39 : 20 85 20 90 02 e6 21 60  d3
1f41 : a2 00 a0 00 ad 00 1b 10  73
1f49 : 04 8e 00 1b 60 a5 8f c9  fd
1f51 : fd 90 06 a5 68 c9 02 d0  4b
1f59 : 03 ee 02 1b a5 8f c9 fe  b3
1f61 : d0 01 88 c9 fd d0 01 c8  09
1f69 : c9 fb d0 01 ca c9 f7 d0  01
1f71 : 01 e8 86 ac 84 ae 20 aa  b1
1f79 : 1f a9 02 8d 12 1b a2 02  28
1f81 : b5 ac f0 16 10 0b b5 a4  05
1f89 : d0 02 d6 a5 d6 a4 4c 9b  c0
1f91 : 1f 18 75 a4 95 a4 90 02  73
1f99 : f6 a5 ca ca 10 e2 20 b8  78
1fa1 : 1e 90 10 a9 32 85 a9 b0  98
1fa9 : 77 a2 03 b5 63 95 a4 ca  f4
1fb1 : 10 f9 60 ce 12 1b d0 c6  7b
1fb9 : 2c 16 1b 10 40 a5 a7 85  94
1fc1 : 1b a5 a4 a6 a5 a4 ac f0  c1
1fc9 : 1d a4 a6 d0 02 c6 1b 88  d0
1fd1 : 20 c2 1e 90 28 c8 d0 02  fc
1fd9 : e6 1b c8 d0 02 e6 1b 20  9d
1fe1 : c2 1e 90 19 b0 3a 18 69  0a
1fe9 : 01 90 01 e8 a4 a6 20 c2  15
1ff1 : 1e 90 0a e9 02 b0 01 ca  56
1ff9 : 20 c2 1e b0 23 a5 6a 29  73
2001 : 08 f0 04 c6 a9 f0 05 a9  e5
2009 : ff 4c df 20 a5 6a 29 02  81
2011 : f0 06 ad 12 d0 29 4f 2c  9e
2019 : a9 00 18 6d 07 1b 85 a9  29
2021 : ad 02 1b c9 02 d0 03 4c  1b
2029 : b8 20 a5 6a 29 04 f0 1f  5d
2031 : a5 8f c9 fd b0 0d a5 25  24
2039 : 38 e5 65 a5 26 e5 66 a9  f0
2041 : 00 f0 12 a5 23 e5 63 a5  2d
2049 : 24 e5 64 a9 01 d0 06 ad  b8
2051 : 12 d0 29 03 24 2a 85 ac  79
2059 : a9 04 85 ae a9 00 8d 02  10
2061 : 1b 20 aa 1f 20 8f 20 b0  7b
2069 : 16 d0 f6 a9 04 85 ae 20  42
2071 : aa 1f 20 94 20 b0 08 d0  8f
2079 : f6 a6 8f a9 ff d0 0b 49  21
2081 : ff 20 df 20 ae 03 1b 8d  17
2089 : 03 1b 8e 01 1b 60 20 89  26
2091 : 1e b0 14 20 89 1e b0 0f  7b
2099 : a6 ac a9 00 38 2a ca 10  20
20a1 : fc c5 8f f0 02 38 60 e6  b3
20a9 : ac a5 ac c9 04 d0 04 a9  b6
20b1 : 00 85 ac c6 ae 18 60 a5  f0
20b9 : 6a 29 01 f0 3c a9 00 8d  42
20c1 : 02 1b a5 8f 49 ff 29 03  eb
20c9 : 08 a5 8f 28 f0 03 49 03  df
20d1 : 2c 49 0c 49 ff 20 df 20  8f
20d9 : 8d 00 1b ad 01 1b 24 6a  31
20e1 : 50 17 c6 aa d0 0d 48 ad  b6
20e9 : 12 d0 85 aa a5 ab 49 ff  f7
20f1 : 85 ab 68 24 ab 30 02 49  c1
20f9 : 10 60 bd 00 dc a8 09 10  00
2101 : c9 ff d0 03 bd 31 1b 20  70
2109 : 2a 21 9d 31 1b 2c 39 1b  80
2111 : 10 20 a8 b5 8e 09 10 49  16
2119 : ff 85 22 98 09 10 49 ff  ad
2121 : 25 22 08 98 28 f0 0b 05  ad
2129 : 22 29 ef 85 22 98 29 10  38
2131 : 05 22 60 78 a9 a8 8d 14  ad
2139 : 03 a9 21 8d 15 03 a9 00  1b
2141 : 8d 1a d0 8d 19 d0 ad 04  98
2149 : 0a 29 fe 8d 04 0a a2 18  a4
2151 : 1e 00 d4 ca 10 fa a9 f0  5f
2159 : 8d 0d d4 a9 b0 8d 06 d4  10
2161 : a9 f0 8d 14 d4 a9 15 8d  72
2169 : 04 d4 a9 11 8d 0b d4 a9  3c
2171 : 9f 8d 18 d4 a9 7f 8d 0d  5e
2179 : dc a9 81 8d 0d dc a9 0e  b6
2181 : 8d 05 dc a2 01 8e 0e dc  93
2189 : 8e 2f 1b ca 8e 2c 1b 8e  a3
2191 : 28 1b 8e 2d 1b 8e 29 1b  91
2199 : 8e 2a 1b a9 1e 8d 01 d4  34
21a1 : a9 f2 8d 17 d4 58 60 d8  4d
21a9 : ae 08 dd d0 0d ee 08 dd  11
21b1 : c6 a8 10 06 ee 08 dd e8  09
21b9 : 86 a8 ad 0d dc a0 01 b9  eb
21c1 : 31 1b 29 ef d9 29 1b f0  fd
21c9 : 08 99 29 1b a9 00 99 2c  a5
21d1 : 1b 88 10 eb ce 2f 1b d0  26
21d9 : 0e a9 0a 8d 2f 1b ee 2c  d0
21e1 : 1b ad 2c 1b 8d 0f d4 ad  41
21e9 : 2d 1b ee 28 1b 10 0b a2  08
21f1 : 00 8e 28 1b 18 69 05 8d  a2
21f9 : 2d 1b 18 69 0f 2d 28 1b  18
2201 : 8d 08 d4 4c 33 ff a2 00  0f
2209 : 2c a2 01 fe 24 1b 86 67  aa
```

Listing 1. »Super-Vectors« ist ein rasantes Action-Spiel, das die hohe Auflösung des C128 im Blechgehäuse nutzt

```
2211 : 20 4b 24 ad 12 d0 a8 29  32
2219 : 0f 85 22 98 4a 4a 4a 4a  3b
2221 : c5 22 f0 ef 98 29 f0 05  d2
2229 : 22 a2 1a 20 cc cd 20 78  d4
2231 : 1e a2 18 1e 00 d4 ca 10  5c
2239 : fa a9 03 8d 0f d4 a9 fc  b3
2241 : 8d 14 d4 a9 f4 8d 17 d4  04
2249 : a9 1f 8d 18 d4 a2 81 8e  6e
2251 : 12 d4 ca 8e 12 d4 a9 10  e0
2259 : 85 22 a0 00 a2 18 20 da  39
2261 : cd 49 40 20 cc cd a2 00  ad
2269 : ca d0 fd 88 d0 ee a2 23  81
2271 : a5 22 49 7f 20 cc cd c6  97
2279 : 22 d0 e1 20 77 2c 20 69  ac
2281 : 29 20 7e 1e 20 2e 24 20  62
2289 : 8b 2c a5 67 c9 04 90 06  8c
2291 : a9 ce a2 24 d0 71 a0 03  f0
2299 : b9 c3 1b 99 bc 24 29 cf  5f
22a1 : 48 88 10 f4 68 f0 08 aa  53
22a9 : a9 00 69 3c ca d0 fb 85  62
22b1 : 22 68 68 0a 8d 21 1b 0a  c5
22b9 : 0a 6d 21 1b 65 22 85 22  e7
22c1 : 68 65 22 a6 67 18 75 41  c9
22c9 : 95 41 90 02 f6 43 a9 b3  fb
22d1 : a2 24 20 cc 2a a5 6a 29  f3
22d9 : 10 f0 17 a9 66 a2 24 20  a9
22e1 : cc 2a a5 67 d0 06 a9 32  61
22e9 : a2 27 d0 14 a9 26 a2 27  7a
22f1 : d0 0e a5 67 d0 06 a9 6c  db
22f9 : a2 24 d0 04 a9 77 a2 24  8b
2301 : 20 cc 2a a9 82 a2 24 20  55
2309 : cc 2a 20 77 2c ce 23 1b  dd
2311 : f0 30 a9 8b a2 24 20 cc  5a
2319 : 2a ad 23 1b c9 01 d0 06  3a
2321 : a9 9f a2 24 d0 04 a9 97  ca
2329 : a2 24 20 cc 2a a9 a6 a2  4f
2331 : 24 20 cc 2a ad 23 1b a2  83
2339 : 18 20 c2 2b 20 2e 24 4c  14
2341 : 6a 1c 20 2e 24 20 57 2c  80
2349 : a9 21 a2 25 20 cc 2a a0  22
2351 : 23 20 7c 2c a0 1f a2 1e  f3
2359 : 20 cc cd 88 d0 fa a2 18  03
2361 : 20 cc cd a9 69 a2 25 20  10
2369 : cc 2a a5 6a 29 10 f0 06  e4
2371 : a9 32 a2 27 d0 04 a9 6c  6d
2379 : a2 24 20 cc 2a ad 24 1b  a6
2381 : a2 23 20 c2 2b a9 6d a2  10
2389 : 25 20 cc 2a a5 6a 29 10  a9
2391 : f0 06 a9 26 a2 27 d0 04  62
2399 : a9 77 a2 24 20 cc 2a ad  97
23a1 : 25 1b a2 39 20 c2 2b a2  2e
23a9 : 1f a0 17 8e c1 1b 8c c2  5d
23b1 : 1b a2 02 8e c3 1b ca 8e  cd
23b9 : c0 1b ca 8e c8 1b 20 da  27
23c1 : 23 a2 35 8e c1 1b a2 01  d6
23c9 : 20 da 23 20 2e 24 68 68  9a
23d1 : 20 7a 2c 20 78 1e 4c 9c  20
23d9 : 26 bd 24 1b d0 0c a0 03  40
23e1 : a9 2d 99 c4 1b 88 10 fa  4c
23e9 : 30 40 a0 00 b5 41 38 fd  a4
23f1 : 24 1b 95 41 b0 0c 49 ff  c1
23f9 : 69 01 85 22 b5 43 f0 05  cc
2401 : d6 43 c8 d0 e7 bd 24 1b  f8
2409 : 4a c5 22 98 69 00 a0 30  4b
2411 : c9 3c 90 05 e9 3c c8 d0  02
2419 : f7 8c c4 1b 20 fb f9 8e  d2
2421 : c6 1b 8d c7 1b a9 3a 8d  d4
2429 : c5 1b 4c 14 2c 20 4b 24  4b
2431 : ad 00 dc 2d 01 dc 29 10  77
2439 : f0 f6 ad 00 dc 2d 01 dc  05
2441 : 29 10 d0 f6 20 92 c9 4c  dc
2449 : 03 c4 78 a9 08 8d 04 d4  a8
2451 : 8d 0b d4 8d 12 d4 a9 00  b9
2459 : 8d 0e dc a2 0c 20 9b ee  87
2461 : ad 0d dc 58 60 03 44 c5  92
2469 : d2 20 00 03 53 d0 c9 c5  1a
2471 : cc c5 d2 20 31 00 03 53  9e
2479 : d0 c9 c5 cc c5 d2 20 32  11
2481 : 00 03 20 d3 c9 c5 c7 d4  19
2489 : c5 00 01 0e 14 03 4e cf  83
2491 : c3 c8 01 22 14 00 03 52  2f
2499 : d5 ce c4 c5 ce 00 03 52  5d
24a1 : d5 ce c4 c5 00 03 20 da  15
24a9 : d5 20 d3 d0 c9 c5 cc c5  27
24b1 : ce 00 01 14 11 03 c9 ce  30
24b9 : 03 02 20 00 00 00 00 20  06
24c1 : 02 03 4d c9 ce d5 d4 c5  4c
24c9 : ce 01 16 0e 00 01 0e 0c  b8
24d1 : 03 44 c9 c5 20 4d c1 d8  46
24d9 : c9 cd c1 cc da c5 c9 d4  3f
24e1 : 20 d6 cf ce 20 34 20 4d  f9
24e9 : c9 ce d5 d4 c5 ce 01 0e  1c
24f1 : 0e c9 d3 d4 20 c5 d2 d2  94
24f9 : c5 c9 c3 c8 d4 2e 01 0e  8c
2501 : 10 44 c9 c5 20 52 d5 ce  e8
2509 : c4 c5 20 d7 c9 d2 c4 20  39
2511 : ce c9 c3 c8 d4 20 c7 c5  c7
2519 : d7 c5 d2 d4 c5 d4 2e 00  de
2521 : 01 02 01 02 2a 2a 2a 20  81
2529 : 45 4e 44 45 52 47 45 42  48
2531 : 4e 49 53 20 2a 2a 2a 01  9b
2539 : 02 02 2d 2d 2d 2d 2d 2d  79
2541 : 2d 2d 2d 2d 2d 2d 2d 2d  41
2549 : 2d 2d 2d 2d 2d 02 01 13  0b
2551 : 0a 03 44 c1 d3 20 53 d0  53
2559 : c9 c5 cc 20 c9 d3 d4 20  0b
2561 : c2 c5 c5 ce c4 c5 d4 00  1f
2569 : 01 1e 10 00 01 34 10 00  6f
2571 : 01 09 14 5a c1 c8 cc 20  1d
2579 : c4 c5 d2 20 53 c9 c5 c7  03
2581 : c5 01 09 17 44 d5 d2 c3  b2
2589 : c8 d3 c3.c8 ce c9 d4 d4  7d
2591 : d3 da c5 c9 d4 00 56 45  ad
2599 : 43 54 4f 52 53 a0 06 b9  ea
25a1 : f0 1b d9 97 25 d0 05 88  86
25a9 : 10 f5 30 64 a0 06 b9 97  9d
25b1 : 25 99 f0 1b 88 10 f7 24  74
25b9 : d7 30 03 20 2c cd a9 50  e6
25c1 : 85 6a a9 80 8d 39 1b a2  4a
25c9 : 07 86 69 20 e4 29 a9 02  b4
25d1 : 85 2a a9 0d 85 2b a9 0a  e4
25d9 : 8d 18 1b 85 27 a9 00 85  b5
25e1 : af 85 5e a9 86 20 59 2c  47
25e9 : a2 0c a9 b3 20 cc cd a9  65
25f1 : 16 a2 2d 20 40 2a a9 71  87
25f9 : a2 25 20 cc 2a 20 7a 2c  b5
2601 : a9 63 a2 2d 20 40 2a 85  62
2609 : 5e a9 60 a2 27 20 cc 2a  a3
2611 : a9 08 8d 1b 1b 20 69 29  30
2619 : 20 77 2c a5 6a 29 10 d0  86
2621 : 06 a9 1e a2 27 d0 04 a9  34
2629 : 2f a2 27 20 cc 2a a5 2a  80
2631 : a2 16 a0 10 20 1b 2c a5  df
2639 : 2b a2 40 a0 10 20 1b 2c  a0
2641 : a2 40 a0 12 a5 27 20 ca  17
2649 : 2b a2 16 a0 14 a5 6a 29  c9
2651 : 80 20 ef 2b a2 40 a0 14  19
2659 : a5 6a 29 20 20 ef 2b a2  f5
2661 : 16 a0 16 ad 18 1b 20 ca  73
2669 : 2b a2 40 a0 16 a5 af 20  97
2671 : ef 2b a5 6a 29 10 d0 09  15
2679 : a9 4c a2 27 20 cc 2a f0  c9
2681 : 10 a9 3c a2 27 20 cc 2a  c4
2689 : a2 40 a0 1a a5 69 20 ca  72
2691 : 2b a2 16 a0 12 ad 39 1b  51
2699 : 20 ef 2b 20 13 27 f0 fb  a6
26a1 : 48 20 92 c9 68 ac 1b 1b  66
26a9 : c9 fe d0 10 ce 1b 1b 10  7a
26b1 : 2b a2 0a a5 6a 29 10 f0  77
26b9 : 20 e8 d0 1d c9 fd f0 06  81
26c1 : 20 02 29 4c 1c 26 ee 1b  9b
26c9 : 1b a2 0b a5 6a 29 10 f0  bf
26d1 : 01 e8 ec 1b 1b d0 05 a2  76
26d9 : 00 8e 1b 1b 98 20 45 2c  43
26e1 : a9 20 8d c3 1b 8d c4 1b  dd
26e9 : a2 01 8e c0 1b ca 8e c5  95
26f1 : 1b 20 14 2c ad 1b 1b 20  07
26f9 : 45 2c a9 3d 8d c3 1b a9  1d
2701 : 3e 8d c4 1b 20 14 2c 20  2e
2709 : 03 c4 20 13 27 d0 fb 4c  5a
2711 : 9c 26 ad 00 dc cd 00 dc  22
2719 : d0 f8 c9 ff 60 01 4e 18  4f
2721 : 20 20 01 40 18 03 53 d0  22
2729 : c9 c5 cc c5 d2 00 01 40  72
2731 : 18 03 43 cf cd d0 d5 d4  fa
2739 : c5 d2 00 01 2d 1a 43 cf  d8
2741 : cd d0 d5 d4 c5 d2 2d 49  c0
2749 : 51 3a 00 01 2d 1a 02 20  c4
2751 : 20 20 20 20 20 20 00 ce  2e
2759 : c5 c9 ce ca c1 20 20 01  af
2761 : 20 00 02 4f 52 42 49 54  f1
2769 : 02 01 23 03 03 d0 d2 c5  a3
2771 : d3 c5 ce d4 d3 03 01 03  d4
2779 : 10 48 c9 ce d4 c5 d2 c7  50
2781 : d2 d5 ce c4 c6 c1 d2 c2  d5
2789 : c5 3a 01 2d 10 56 cf d2  ea
2791 : c4 c5 d2 c7 d2 d5 ce c4  86
2799 : c6 c1 d2 c2 c5 3a 01 03  85
27a1 : 12 44 c9 c1 c7 cf ce c1  3a
27a9 : cc c5 ce 3a 01 2d 12 47  a3
27b1 : c5 d3 c3 c8 d7 c9 ce c4  fa
27b9 : c9 c7 cb c5 c9 d4 3a 01  3f
27c1 : 03 14 42 c5 d3 c3 c8 cc  30
27c9 : c5 d5 ce c9 c7 d5 ce c7  5b
27d1 : 3a 01 2d 14 48 c9 ce c4  f1
27d9 : c5 d2 ce c9 d3 d3 c5 3a  5b
27e1 : 01 03 16 52 d5 ce c4 c5  a6
27e9 : ce da c1 c8 cc 3a 01 2d  ab
27f1 : 16 5a d5 c6 c1 cc cc d3  e0
27f9 : d3 d4 c1 d2 d4 3a 01 2d  7e
2801 : 18 47 c5 c7 ce c5 d2 3a  02
2809 : 01 00 18 3d 3e 20 02 53  4c
2811 : 54 41 52 54 01 03 1a 45  40
2819 : 4e 44 45 02 01 00 1d 41  22
2821 : d2 ce c5 20 4a c1 ce d3  65
2829 : c5 ce 20 26 20 48 c1 d2  13
2831 : c1 cc c4 20 42 c5 c9 ce  a5
2839 : c5 20 31 39 38 36 20 2f  96
2841 : 20 46 d2 c1 ce cb 20 48  cd
2849 : c5 c9 c4 c5 d2 cd c1 ce  1d
2851 : ce d3 20 31 39 38 38 20  ae
2859 : 2d 20 28 43 29 20 43 cf  49
2861 : d0 d9 d2 c9 c7 c8 d4 20  62
2869 : c2 d9 20 36 34 27 c5 d2  20
2871 : 00 d3 c3 c8 d7 c1 d2 da  f1
2879 : 20 20 c7 d2 c1 d5 20 20  81
2881 : 20 20 20 c2 cc c1 d5 20  84
2889 : 20 20 20 20 c8 c5 cc cc  4d
2891 : c2 cc c1 d5 20 c7 d2 d5  1c
2899 : c5 ce 20 20 20 20 c8 c5  83
28a1 : cc cc c7 d2 d5 c5 ce c2  6c
28a9 : cc c1 d5 c7 d2 c1 d5 20  97
28b1 : d4 d5 c5 d2 cb c9 d3 20  d6
28b9 : 20 c4 d5 ce cb c5 cc d2  4e
28c1 : cf d4 c8 c5 cc cc d2 cf  03
28c9 : d4 20 20 d6 c9 cf cc c5  6a
28d1 : d4 d4 20 20 d2 cf d3 c1  9a
28d9 : 20 20 20 20 20 c2 d2 c1  fc
28e1 : d5 ce 20 20 20 20 c7 c5  d7
28e9 : cc c2 20 20 20 20 20 c8  38
28f1 : c5 cc cc c7 d2 c1 d5 20  1b
28f9 : d7 c5 c9 d3 d3 20 20 20  9f
```

```
2901 : 20 8d 20 1b ad 1b 1b 0a   88
2909 : aa bd 19 29 bc 1a 29 8d   5a
2911 : 21 1b 8c 22 1b 6c 21 1b   f7
2919 : 31 29 34 29 b8 29 76 29   12
2921 : 8f 29 92 29 9c 29 c1 29   7b
2929 : 41 1c 95 29 02 2a c8 29   ea
2931 : a2 00 2c a2 01 ac 20 1b   5f
2939 : a9 0f c0 fb d0 06 d6 2a   06
2941 : 30 1e 10 1e c0 f7 d0 08   67
2949 : d5 2a f0 12 f6 2a d0 12   da
2951 : c0 ef d0 39 a5 2a a6 2b   01
2959 : 85 2b 86 2a b0 0a a9 00   5d
2961 : 95 2a a5 2a c5 2b f0 d0   d5
2969 : a5 2b 0a 0a 0a 0a 05 2a   c1
2971 : a2 1a 4c cc cd ad 20 1b   ce
2979 : c9 fb d0 07 c6 27 d0 0d   58
2981 : e6 27 60 c9 f7 d0 06 a5   b5
2989 : 27 c9 14 d0 f3 60 a9 80   9e
2991 : 2c a9 20 2c a9 10 45 6a   24
2999 : 85 6a 60 ad 20 1b c9 fb   1b
29a1 : d0 09 ce 18 1b d0 0f ee   ff
29a9 : 18 1b 60 c9 f7 d0 07 ad   1e
29b1 : 18 1b c9 63 d0 f1 60 ad   af
29b9 : 39 1b 49 80 8d 39 1b 60   b2
29c1 : a5 af 49 80 85 af 60 ad   53
29c9 : 20 1b c9 fb d0 07 c6 69   9c
29d1 : d0 0f e6 69 60 c9 f7 d0   e5
29d9 : 08 a5 69 c9 0a f0 02 e6   45
29e1 : 69 a6 69 bd 1d 2a 48 29   46
29e9 : 0f 85 22 68 29 f0 8d 16   cd
29f1 : 1b 29 7f 09 01 8d 07 1b   71
29f9 : a5 6a 29 f0 05 22 85 6a   88
2a01 : 60 ad 20 1b c9 ef d0 f8   f5
2a09 : 8d f0 1b 20 09 e1 20 00   f9
2a11 : c0 20 0c ce 20 b6 77 ee   32
2a19 : 04 0a 4c 03 40 00 01 05   a8
2a21 : 8d 8f af ff 8a 81 0a 01   40
2a29 : 02 03 04 05 06 07 08 0a   1b
2a31 : 0c 0f 12 16 1a 1e 23 28   7c
2a39 : 2d 32 3c fe fd fb f7 85   19
2a41 : 5c 86 5d a0 00 b1 5c 85   56
2a49 : a4 c8 b1 5c 10 01 60 85   df
2a51 : a5 c8 b1 5c 85 a6 a5 5e   33
2a59 : f0 03 c8 b1 5c 85 1b c8   23
2a61 : b1 5c 85 a3 c8 b1 5c d0   43
2a69 : 11 20 c0 2a 20 c4 1e 69   73
2a71 : 00 d0 01 e8 c6 a3 d0 f4   ed
2a79 : f0 c9 c9 01 d0 11 20 c0   78
2a81 : 2a 20 c4 1e c8 d0 02 e6   99
2a89 : 1b c6 a3 d0 f4 f0 b4 c9   47
2a91 : 02 d0 16 20 c0 2a 20 c4   ec
2a99 : 1e e9 01 b0 01 ca c8 d0   2d
2aa1 : 02 e6 1b c6 a3 d0 ef f0   18
2aa9 : 9a 20 c0 2a 20 c4 1e 69   3c
2ab1 : 00 d0 01 e8 c8 d0 02 e6   60
2ab9 : 1b c6 a3 d0 ef f0 84 c8   64
2ac1 : 98 20 84 2b a5 a4 a6 a5   55
2ac9 : a4 a6 60 85 5c 86 5d a0   3a
2ad1 : 00 8c 10 1b b1 5c f0 f2   26
2ad9 : c8 8c 0e 1b c9 01 d0 11   d8
2ae1 : b1 5c 8d 09 1b c8 b1 5c   bc
2ae9 : 0a 0a 0a 8d 0a 1b c8 d0   6b
2af1 : e3 c9 03 f0 0a c9 02 d0   30
2af9 : 1a a9 03 a2 00 f0 04 a9   e8
2b01 : 01 a2 80 8d 11 1b 8e 21   8c
2b09 : 1b ad 10 1b 49 ff 8d 10   4d
2b11 : 1b b0 c1 c9 c0 90 02 e9   9a
2b19 : c0 a2 00 86 5b 86 22 a0   af
2b21 : 03 0a 26 5b 88 d0 fa 85   24
2b29 : 5a a5 5b 69 d8 85 5b ad   dc
2b31 : 09 1b a2 03 0a 26 22 ca   c1
2b39 : d0 fa 86 1b a6 22 ac 0a   ce
2b41 : 1b 20 c2 1e a0 00 20 2a   c0
2b49 : 1f ee 00 ff b1 5a ce 00   08
2b51 : ff 2c 10 1b 10 03 20 8e   85
2b59 : 2b 20 ca cd a9 60 20 37   8d
2b61 : 1f c8 c0 08 d0 e0 2c 10   fa
2b69 : 1b 10 0a ad 09 1b 18 6d   69
2b71 : 11 1b 8d 09 1b ee 09 1b   18
2b79 : ad 0e 1b 20 84 2b a0 00   1c
2b81 : 4c d5 2a 18 65 5c 85 5c   4d
2b89 : 90 02 e6 5d 60 8c 12 1b   68
2b91 : a2 08 0a 08 ac 11 1b 28   cb
2b99 : 08 2c 21 1b 30 04 26 62   e4
2ba1 : 26 61 26 60 26 5f 88 10   ad
2ba9 : ee 28 ca d0 e5 c8 b9 5f   c2
2bb1 : 00 20 ca cd c8 cc 11 1b   9b
2bb9 : d0 f4 b9 5f 00 ac 12 1b   41
2bc1 : 60 a0 14 c9 0a b0 02 e8   b0
2bc9 : e8 8e c1 1b 8c c2 1b a2   5d
2bd1 : 00 8e c6 1b e8 8e c0 1b   6a
2bd9 : e8 8e c3 1b 20 fb f9 e0   e8
2be1 : 30 d0 03 aa a9 20 8e c4   ef
2be9 : 1b 8d c5 1b d0 25 8e c1   94
2bf1 : 1b 8c c2 1b a2 03 8e c3   6a
2bf9 : 1b a2 01 8e c0 1b ca 8e   a5
2c01 : c8 1b a8 f0 02 a0 04 b9   48
2c09 : 58 27 9d c4 1b c8 e8 e0   52
2c11 : 04 d0 f4 a9 c0 a2 1b 4c   16
2c19 : cc 2a 8e c1 1b 8c c2 1b   2d
2c21 : a2 03 8e c3 1b a2 01 8e   49
2c29 : c0 1b ca 8e cd 1b 85 22   0b
2c31 : 0a 0a 0a 65 22 a8 b9 72   a3
2c39 : 28 9d c4 1b c8 e8 e0 09   2e
2c41 : d0 f4 f0 cf 4a 90 03 a2   3c
2c49 : 2a 2c a2 00 8e c1 1b 0a   a9
2c51 : 69 10 8d c2 1b 60 a9 00   d9
2c59 : a0 85 a2 12 20 cc cd e8   18
2c61 : a9 00 20 cc cd 20 ca cd   51
2c69 : ca 20 cc cd 88 d0 fa 60   ec
2c71 : 20 57 2c a0 0b 2c a0 10   11
2c79 : 2c a0 08 be 92 2c 30 ef   fa
2c81 : c8 b9 92 2c 20 cc cd c8   81
2c89 : d0 f1 20 71 2c a0 15 d0   46
2c91 : ea 01 60 04 2e 05 01 1c   dc
2c99 : ff 0c 86 ff 22 70 23 78   63
2ca1 : ff 22 7d 23 64 ff 02 67   92
2ca9 : 06 1e 07 27 ff 02 6d 06   37
2cb1 : 2e 07 2d ff 12 2d 13 00   85
2cb9 : 18 80 20 e0 21 00 ff 00   48
2cc1 : 00 00 00 00 00 00 01 00   c6
2cc9 : 00 5d 00 5c 01 01 00 07   2a
2cd1 : 01 5c 01 08 00 28 00 83   8a
2cd9 : 01 01 00 07 01 83 01 00   6c
2ce1 : 00 00 00 83 02 00 00 5e   2f
2ce9 : 00 00 00 60 01 00 00 00   06
2cf1 : 01 60 01 00 00 00 02 60   2c
2cf9 : 01 e0 00 00 00 01 00 00   73
2d01 : 01 00 00 01 01 5f 01 e0   f3
2d09 : 02 01 00 00 01 e0 02 01   ad
2d11 : 01 5f 01 ff ff 3b 00 78   cd
2d19 : 00 00 3b 01 78 00 00 3b   06
2d21 : 00 90 00 00 3b 01 90 00   67
2d29 : 00 3b 00 94 00 00 3b 01   48
2d31 : 94 00 00 3b 00 ae 00 00   a2
2d39 : 3b 01 ae 00 00 3b 00 c8   0c
2d41 : 00 00 3b 01 c8 00 00 3a   31
2d49 : 00 78 51 01 de 00 79 50   6e
2d51 : 01 e4 00 79 50 01 90 01   45
2d59 : 79 50 01 3a 02 78 51 01   ad
2d61 : ff ff 00 00 00 64 03 c8   21
2d69 : 00 00 64 02 00 00 00 05   cd
2d71 : 00 05 00 00 5f 03 c3 00   11
2d79 : 00 5f 02 be 00 00 05 00   95
2d81 : be 00 00 5a 02 0a 00 00   fb
2d89 : 5a 03 9b 00 32 32 01 9b   3c
2d91 : 00 64 05 00 a0 00 32 32   3c
2d99 : 01 a0 00 32 05 00 a5 00   18
2da1 : 32 32 01 a5 00 64 46 00   1e
2da9 : eb 00 62 03 01 aa 00 62   b8
2db1 : 41 00 aa 00 60 03 01 aa   14
2db9 : 00 60 41 00 eb 00 5e 03   78
2dc1 : 01 aa 00 5e 41 00 aa 00   a2
2dc9 : 4e 10 01 aa 00 4e 28 00   c8
2dd1 : d2 00 4c 03 01 aa 00 4c   15
2dd9 : 28 00 aa 00 4a 03 01 aa   c2
2de1 : 00 4a 28 00 d2 00 48 03   65
2de9 : 01 aa 00 48 28 00 aa 00   76
2df1 : 38 10 01 aa 00 38 41 00   8e
2df9 : eb 00 36 03 01 aa 00 36   a4
2e01 : 41 00 aa 00 34 03 01 aa   a2
2e09 : 00 34 41 00 eb 00 32 03   01
2e11 : 01 aa 00 32 41 00 f0 00   86
2e19 : 32 46 00 f0 00 32 32 01   e9
2e21 : f0 00 64 46 00 36 01 62   6e
2e29 : 03 01 f5 00 62 41 00 f5   46
2e31 : 00 34 2e 01 f5 00 34 41   aa
2e39 : 00 36 01 34 03 01 fa 00   3f
2e41 : 36 3c 00 fa 00 36 2a 01   51
2e49 : fa 00 60 3c 00 3b 01 32   25
2e51 : 46 00 81 01 32 03 01 3b   ce
2e59 : 01 34 46 00 3b 01 34 03   99
2e61 : 01 3b 01 36 46 00 81 01   73
2e69 : 36 03 01 65 01 38 1b 00   4c
2e71 : 65 01 38 2c 01 60 01 64   ca
2e79 : 05 00 60 01 38 2c 01 5b   56
2e81 : 01 38 05 00 5b 01 38 2c   d7
2e89 : 01 56 01 64 05 00 56 01   2e
2e91 : 38 2c 01 86 01 32 46 00   ab
2e99 : cc 01 32 32 01 86 01 64   ca
2ea1 : 46 00 86 01 34 30 01 86   7f
2ea9 : 01 34 41 00 c7 01 34 2e   c6
2eb1 : 01 8b 01 62 3c 00 8b 01   f8
2eb9 : 36 2c 01 8b 01 36 37 00   56
2ec1 : c2 01 36 2a 01 90 01 60   30
2ec9 : 32 00 90 01 38 28 01 90   2a
2ed1 : 01 38 2d 00 d6 01 32 32   dc
2ed9 : 01 d6 01 64 05 00 db 01   d4
2ee1 : 32 32 01 db 01 32 05 00   9e
2ee9 : e0 01 32 32 01 e0 01 64   01
2ef1 : 05 00 e5 01 36 2e 01 e5   34
2ef9 : 01 36 23 00 08 02 36 0f   66
2f01 : 01 ea 01 45 1e 00 ea 01   f0
2f09 : 45 03 01 ea 01 47 23 00   44
2f11 : 0d 02 34 14 01 e5 01 34   5b
2f19 : 28 00 e5 01 32 03 01 e5   e6
2f21 : 01 32 2d 00 12 02 32 18   b1
2f29 : 01 ea 01 49 28 00 ea 01   39
2f31 : 49 03 01 ea 01 4b 0f 00   40
2f39 : f9 01 4b 19 01 f9 01 64   56
2f41 : 05 00 fe 01 4b 05 00 fe   01
2f49 : 01 4b 19 01 03 02 4b 1a   f8
2f51 : 01 1c 02 32 46 00 1c 02   00
2f59 : 32 03 01 1c 02 34 46 00   ac
2f61 : 62 02 34 03 01 1c 02 36   97
2f69 : 46 00 1c 02 36 17 01 1c   4f
2f71 : 02 4d 05 00 21 02 38 15   88
2f79 : 01 21 02 38 05 00 26 02   7f
2f81 : 38 15 01 26 02 4d 37 00   b0
2f89 : 5d 02 4b 03 01 2b 02 4b   23
2f91 : 32 00 2b 02 49 03 01 2b   d6
2f99 : 02 49 37 00 62 02 49 1b   9f
2fa1 : 01 5d 02 64 05 00 5d 02   28
2fa9 : 4f 15 01 58 02 4f 05 00   7d
2fb1 : 58 02 4f 15 01 1c 02 64   43
2fb9 : 3c 00 1c 02 62 03 01 1c   b7
2fc1 : 02 62 37 00 53 02 60 03   8f
2fc9 : 01 1c 02 60 37 00 00 00   d8
2fd1 : 69 00 00 00 01 69 00 00   96
```

Listing 1. »Super-Vectors« (Fortsetzung)

```
2fd9 : 00 02 69 80 00 00 00 6b  1c
2fe1 : 00 00 00 01 6b 00 00 00  b8
2fe9 : 02 6b 80 00 00 00 6d 00  77
2ff1 : 00 00 01 6d 00 00 00 02  e3
2ff9 : 6d 80 00 7f 02 69 03 01  10
3001 : 00 00 6b 03 01 9f 01 00  4e
3009 : 05 00 a1 01 00 03 01 a5  fe
3011 : 01 00 03 01 a6 01 00 02  6a
3019 : 03 a8 01 00 02 02 a9 01  8a
3021 : 00 03 01 ff ff 46 30 84  df
3029 : 30 ce 30 4e 31 92 31 ac  5c
3031 : 31 02 32 28 32 8a 32 40  b6
3039 : 33 00 00 00 00 00 00 00  6d
3041 : 00 00 00 00 00 64 00 32  c9
3049 : 00 ff 01 7b 02 32 00 ff  aa
3051 : 01 d0 00 51 00 ff 00 cf  84
3059 : 01 51 00 41 00 d0 00 15  dc
3061 : 01 ff 00 cf 01 15 01 41  9b
3069 : 00 52 01 9d 00 28 01 52  70
3071 : 01 9d 00 3c 00 8e 01 9d  7c
3079 : 00 28 01 52 01 c5 00 3d  d1
3081 : 00 ff ff e9 00 5f 00 e4  83
3089 : 00 e9 00 5f 00 2a 03 cd  63
3091 : 01 5f 00 2a 03 13 01 89  67
3099 : 00 e4 00 e9 00 5f 00 7e  41
30a1 : 01 13 01 89 00 7e 01 f7  85
30a9 : 01 89 00 7f 01 e9 00 dd  7a
30b1 : 00 2a 03 13 01 07 01 e4  00
30b9 : 00 cd 01 5f 00 7e 01 cd  60
30c1 : 01 dd 00 2a 03 e9 00 dd  31
30c9 : 00 e4 00 ff ff f8 00 94  2c
30d1 : 00 32 00 f8 00 94 00 14  d6
30d9 : 02 e4 00 a8 00 19 01 e4  f9
30e1 : 00 c1 00 14 03 f8 00 d5  e8
30e9 : 00 32 00 3e 01 c1 00 15  13
30f1 : 02 2a 01 ad 00 14 03 f8  9d
30f9 : 00 ad 00 32 00 f8 00 ad  39
3101 : 00 14 02 5c 01 94 00 2d  27
3109 : 01 5c 01 c1 00 4b 00 8e  28
3111 : 01 a8 00 2d 01 c0 01 94  4f
3119 : 00 14 02 d4 01 c6 00 28  d5
3121 : 00 f2 01 bc 00 0a 03 de  8c
3129 : 01 bc 00 0a 02 de 01 bc  5e
3131 : 00 14 00 d4 01 c6 00 0f  3a
3139 : 03 e3 01 d5 00 14 00 10  ea
3141 : 02 bc 00 19 01 1a 02 bc  27
3149 : 00 0a 02 ff ff 70 01 b1  ba
3151 : 00 54 03 70 01 b1 00 54  91
3159 : 02 1c 01 05 01 a9 00 46  34
3161 : 00 3a 00 78 00 46 00 3a  34
3169 : 00 50 01 50 00 94 00 64  49
3171 : 00 be 00 3a 00 50 01 22  e3
3179 : 02 3a 00 78 00 22 02 3a  35
3181 : 00 50 01 9a 02 3a 00 50  d0
3189 : 01 2c 02 94 00 64 00 ff  d7
3191 : ff 55 01 41 00 e0 02 8a  c7
3199 : 01 41 00 e0 03 55 01 79  29
31a1 : 00 70 02 8a 01 79 00 70  68
31a9 : 03 ff ff 00 00 af 00 ff  29
31b1 : 00 ff 00 af 00 67 00 00  e2
31b9 : 00 b3 00 ff 00 ff 00 b3  fa
31c1 : 00 67 00 7a 01 af 00 ff  52
31c9 : 00 79 02 af 00 67 00 7a  2d
31d1 : 01 b3 00 ff 00 79 02 b3  e7
31d9 : 00 67 00 34 01 a5 00 78  42
31e1 : 00 34 01 bd 00 78 00 8c  d0
31e9 : 00 0e 00 78 02 54 02 0e  e6
31f1 : 00 78 03 14 00 dc 00 78  49
31f9 : 03 cc 02 dc 00 78 02 ff  4a
3201 : ff 04 00 6b 00 c4 00 00  96
3209 : 00 f7 00 c9 00 c8 00 6b  5b
3211 : 00 8c 01 17 02 6b 00 c5  82
3219 : 00 17 02 f7 00 c9 00 17  a1
3221 : 02 6b 00 8c 01 ff ff 66  47
3229 : 01 43 00 dc 01 7a 01 43  d6
3231 : 00 dc 01 66 01 43 00 14  ff
3239 : 00 66 01 1f 01 15 00 48  da
3241 : 01 43 00 14 00 48 01 43  33
3249 : 00 6e 01 84 01 43 00 14  a4
3251 : 00 84 01 43 00 6e 01 5c  6c
3259 : 01 43 00 6e 01 98 01 43  29
3261 : 00 6e 01 98 01 b1 00 60  4a
3269 : 03 84 01 b1 00 6b 03 48  1d
3271 : 01 b1 00 60 02 5c 01 b1  c1
3279 : 00 6b 02 e8 00 11 01 0a  6d
3281 : 03 f8 01 11 01 0a 02 ff  cb
3289 : ff c8 00 61 00 a3 00 75  21
3291 : 01 61 00 a3 00 c8 00 01  00
3299 : 01 a3 00 75 01 01 01 a3  7e
32a1 : 00 d2 00 b1 00 ff 00 d1  e4
32a9 : 01 b1 00 3e 00 c8 00 01  93
32b1 : 00 00 01 18 02 61 00 ff  20
32b9 : 01 00 00 bb 00 a0 00 40  b7
32c1 : 02 a7 00 a0 00 a0 00 39  22
32c9 : 00 82 01 40 02 a7 00 82  b5
32d1 : 01 14 00 61 00 8c 00 40  ee
32d9 : 02 01 01 8c 00 14 00 39  41
32e1 : 00 8c 00 00 00 4d 00 8c  ab
32e9 : 00 14 00 61 00 8c 00 00  84
32f1 : 00 75 00 8c 00 14 00 89  f1
32f9 : 00 8c 00 00 00 9d 00 8c  46
3301 : 00 40 02 29 01 8c 00 54  e4
3309 : 02 15 01 8c 00 40 02 01  74
3311 : 01 8c 00 54 02 ed 00 8c  8c
3319 : 00 40 02 d9 00 8c 00 54  02
3321 : 02 c5 00 8c 00 a0 00 bb  14
3329 : 00 50 02 c8 00 b1 00 64  41
3331 : 02 7c 02 4d 00 64 02 90  e8
3339 : 02 57 00 50 02 ff ff 28  61
3341 : 00 a7 00 ff 00 27 01 a7  a1
3349 : 00 ff 00 26 02 a7 00 91  8e
3351 : 00 28 00 bb 00 ff 00 27  2b
3359 : 01 bb 00 ff 00 26 02 bb  e9
3361 : 00 91 00 28 00 43 00 ff  49
3369 : 00 27 01 43 00 ff 00 26  f2
3371 : 02 43 00 91 00 28 00 1f  c7
3379 : 01 ff 00 27 01 1f 01 ff  6c
3381 : 00 26 02 1f 01 91 00 28  e6
3389 : 00 bb 00 64 01 28 00 43  cb
3391 : 00 64 01 b6 02 43 00 64  de
3399 : 01 b6 02 bb 00 64 01 ff  15
33a1 : ff 00 ff 00 ff 00 ff 00  a0
```

Listing 1. »Super-Vectors« (Schluß)

SOLITAIRE

Ein großes Problem für jeden C 128-Fan ist immer noch der Mangel an guten Spielen. Einen Beitrag, um diesen Mangel zu beheben, liefern wir mit Solitaire, einer gelungenen Umsetzung des bekannten Brettspiels auf den C 128

Obwohl das Basic 7.0 des C 128 eine große Anzahl von Befehlen bietet, um die Spieleprogrammierung auf diesem Computer zu vereinfachen, fehlen noch immer gute Programme. Solitaire nutzt die Grafikeigenschaften dieses Computers für eine einfache aber sehr gelungene Umsetzung des bekannten Brettspiels.

Solitaire ermöglicht eine sehr einfache Bedienung per Joystick und Tastatur. Es wird mit »BLOAD'solitaire 128'« geladen und mit »RUN« gestartet. Nach dem Start zeichnet das Programm den Bildschirm mit dem Spielfeld und den 32 Kugeln (Bild 1). Diese können mit dem Joystick bewegt werden. Dazu wird die gewünschte Kugel mit einem Druck auf den Feuerknopf angeklickt, worauf sie ihre Farbe ändert. Beim nächsten Druck auf den Feuerknopf wird die Kugel auf das Feld bewegt, auf das Sie den Rahmen verschoben haben. Die dabei übersprungene Kugel verschwindet vom Spielfeld.

Solitaire erkennt jeden Verstoß gegen die Spielregeln. Falls der Spieler versucht, seine Kugeln auf Felder zu setzen, die nicht erlaubt sind, wird der Spielzug einfach ignoriert. Haben Sie versehentlich eine falsche Kugel angewählt, so signalisieren Sie durch nochmaliges Anklicken

derselben Kugel, daß dies rückgängig gemacht werden soll. Daraufhin wird die Markierung gelöscht und die nächste Kugel kann angeklickt werden.

War der Zug dagegen korrekt (bei jedem Zug muß eine Kugel übersprungen werden, dazu wird ein freier Platz auf

Schummeln unmöglich

der gegenüberliegenden Seite benötigt), so zeigt Solitaire den neuen Spielfeldaufbau an. Man überspringt auf diese Weise eine Kugel nach der anderen, um sie aus dem Spielfeld zu löschen. Das Ziel ist es, so lange zu springen, bis nur noch genau eine Kugel übrig ist. Um dies zu erreichen, ist es nötig, eine gewisse Strategie zu verfolgen:

Als erstes räumen Sie drei beliebige Ecken frei. In der vierten Ecke sollten Sie die letzten vier Kugeln in eine L-förmige Anordnung bringen, bevor Sie Ihre letzten drei Züge anwenden: Als erstes springen Sie auf der kurzen Seite Ihres »L«, dann mit der langen auf die freigewordene Stelle und von dort über die letzte Kugel. Natürlich gibt es einige Varianten, die Sie noch ausführlich ausprobieren können, wenn das Programm (Listing 1) eingegeben ist.

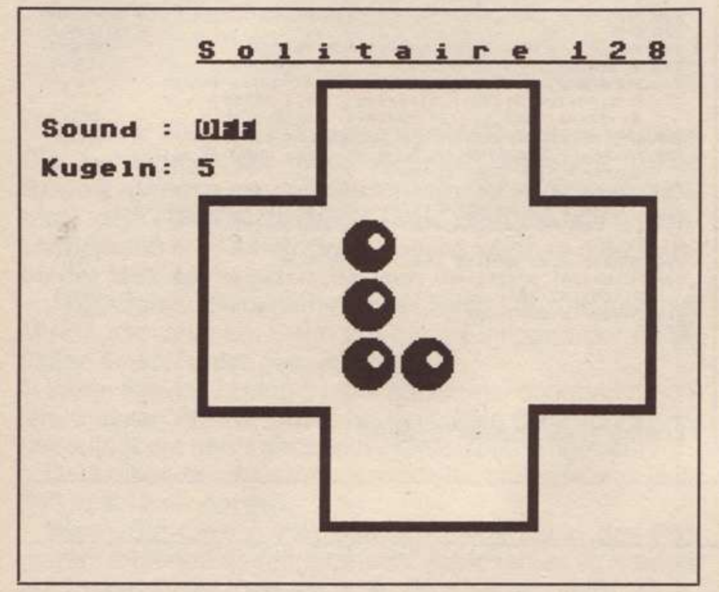


Bild 1. Diese Stellung sollten Sie anstreben, um Solitaire zu lösen

Ganz so einfach, wie diese Beschreibung klingt, ist Solitaire allerdings auch nicht zu lösen. Aber auch für den Fall, daß Sie irgendwann merken sollten, Sie kommen auf dem eingeschlagenen Weg nicht weiter, hat Solitaire vorgesorgt:

Es gibt eine eingebaute Hilfsfunktion, die es Ihnen erlaubt, Ihren Weg einfach ein Stück zurückzugehen, um

Nützliche Hilfsfunktion

dann einen neuen zu versuchen. Drücken Sie dazu die beiden Tasten <SHIFT> und <CTRL> gleichzeitig. Sie sind damit genau um einen Spielzug zurückversetzt.

Wollen Sie dagegen das Spiel neu beginnen, so drücken Sie gleichzeitig die drei Tasten <CTRL>, <SHIFT> und <CBM>. Die Ausgangsstellung von Solitaire wird daraufhin aufgebaut, Sie können nochmals starten.

Wem das Programm nicht komplex genug ist, der kann sich durchaus seine eigene Erweiterung bauen. Beispielsweise könnte man die Zahl der Kugeln (im Moment 32) erhöhen, oder als zusätzliche Spielregel die Bedingung einführen, daß die letzte Kugel in der Mitte landen muß. Sie müssen lediglich darauf achten, daß Sie Ihre Erweiterungen nicht nur in die Grafik einbauen, sondern auch abfragen. Solitaire ist so übersichtlich programmiert, daß Erweiterungen problemlos möglich sind.

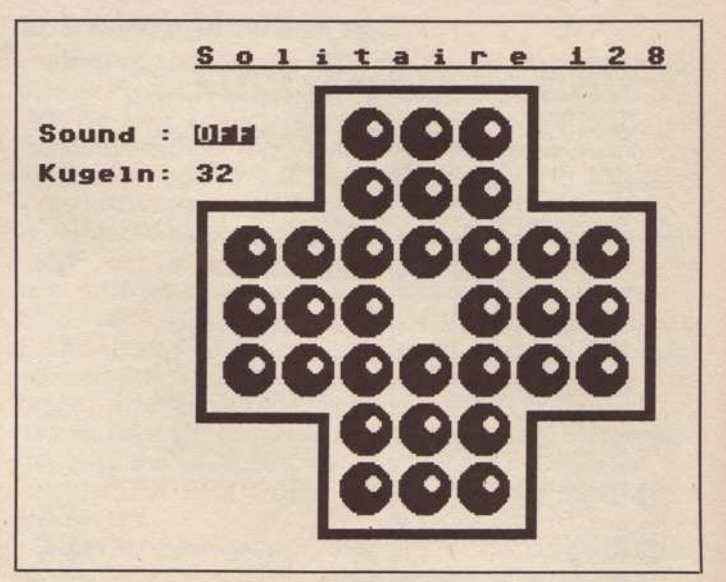


Bild 2. Dies ist die Ausgangsstellung von Solitaire

Eingabehinweise

Bitte geben Sie das Programm mit dem Checksummer (Seite 158) ein und speichern Sie es auf Diskette. Danach können Sie es mit

```
BLOAD "SOLITAIRE 128"
```

laden und mit RUN starten. (Achim Schlieper/so)

```
1 REM *************************************        <640>
2 REM * SOLITAIRE 128                     *        <68P>
3 REM *                                   *        <6CP>
4 REM * GESCHRIEBEN VON : ACHIM SCHLIEPER *        <BHE>
5 REM *                   ARDEYSTR.239    *        <6KQ>
6 REM *                   5810 WITTEN 6   *        <6OR>
7 REM *************************************        <6SR>
10 CLR                                              <ALB>
20 DIM FE(9,9): DIM S1(33): DIM S2(33): DIM D1(33
   ): DIM D2(33): DIM M1(33): DIM M2(33): GOTO 10
   0                                                <UK0>
30 REM --- UNTERPROGRAMM ZEIGER BEWEGEN ---         <APB>
40 MOVSPR 8,75+(24*X1),51+(24*Y1): RETURN           <6I4>
50 REM --- SPRITE 2LOESCHEN+SOUND ---               <B9D>
60 EN=0: SPRITE 2,0: SOUND 1,500,SO: FOR L=1 TO 3
   0: NEXT L: RETURN                                <LP6>
70 --- SOUND OFF /ON ---                            <IIH>
80 COLOR 1,14: IF SO=10 THEN SO=0: CHAR ,8,4,"OFF
   ",1: FOR U=1 TO 30: NEXT : GOTO 330: ELSE SO=1
   0: CHAR ,8,4,"ON ",1: FOR U=1 TO 30: NEXT : GO
   TO 330                                           <L8F>
90 REM --- SPIELFELDAUFBAU ---                      <891>
100 RESTORE : FOR O=1 TO 8: SPRITE O,0: NEXT O      <JG2>
110 COLOR 0,1: COLOR 4,1: COLOR 1,8: WIDTH 1: GRA
    PHIC 1,1: FAST                                  <PEL>
120 CIRCLE 1,20,20,10: CIRCLE 1,23,18,4: PAINT 1,
    18,22: SSHAPE A$,8,10,31,31: SPRSAV A$,1: GRA
    PHIC 1,1: BOX 1,8,10,30,30: SSHAPE B$,8,10,31
    ,31: SPRSAV B$,8: GRAPHIC 1,1                   <11D>
130 FOR Y=0 TO 1: FOR X=1 TO 3: GSHAPE A$,98+(24*
    X),25+(24*Y),0: NEXT X,Y                        <GB0>
140 FOR Y=2 TO 4: FOR X=1 TO 7: GSHAPE A$,50+(24*
    X),25+(24*Y),0: NEXT X,Y: GSHAPE A$,50+(24*4)
    ,25+(24*3),4: SPRSAV A$,2: SPRSAV A$,3: SPRSA
    V A$,4                                          <KU0>
150 FOR Y=5 TO 6: FOR X=1 TO 3: GSHAPE A$,98+(24*
    X),25+(24*Y),0: NEXT X,Y                        <TIF>
160 COLOR 1,16                                      <CSM>
170 DRAW 1,116,18 TO 200,18 TO 200,66 TO 248,66 T
    O 248,148 TO 200,148 TO 200,196 TO 118,196 TO
     118,149 TO 68,149 TO 68,70 TO 68,66 TO 116,6
    6 TO 116,18                                     <TCQ>
180 DRAW 1,113,15 TO 203,15 TO 203,63 TO 251,63 T
    O 251,151 TO 203,151 TO 203,199 TO 115,199 TO
     115,152 TO 65,152 TO 65,67 TO 65,63 TO 113,6
    3 TO 113,15                                     <VJ8>
```

Listing 1. Bitte mit dem Checksummer (Seite 158) eingeben

```
190 PAINT 1,120,16                                      <D7P>
200 COLOR 1,2: CHAR ,8,0,"(CTRL+N)S O L I T A I R
    E(2SPACE)1 2 8": CHAR ,8,1,"TTTTTTTTTTTTTTTT
    TTTTTTTT"                                           <U39>
210 COLOR 1,14: CHAR ,0,4,"(CTRL+N)SOUND :": CHAR
    ,8,4,"OFF",1                                        <SM2>
220 CHAR ,0,6,"(CTRL+N)KUGELN:": CHAR ,8,6,"32"         <261>
230 REM --- ZEIGER POSITIONIEREN ---                    <JQF>
240 MOVSPR 8,75+(24*4),51+(24*4): SPRITE 8,1,16         <CLE>
250 REM --- DATAZEILEN + EINLESEN ---                   <IA9>
260 V(1)=1: V(5)=2: V(3)=3: V(7)=4: T(7)=1: T(5)=
    2: T(2)=3                                           <GDG>
270 X1=4: Y1=4: X0=0: Y0=0: XM=4: YM=4: ZA=32           <JNS>
280 DATA 2,2,2,2,2,2,2,2,2: DATA 2,2,2,1,1,1,2,2,
    2: DATA 2,2,2,1,1,1,2,2,2                           <0MV>
290 DATA 2,1,1,1,1,1,1,1,2: DATA 2,1,1,1,0,1,1,1,
    2: DATA 2,1,1,1,1,1,1,1,2                           <HP2>
300 DATA 2,2,2,1,1,1,2,2,2: DATA 2,2,2,1,1,1,2,2,
    2: DATA 2,2,2,2,2,2,2,2,2                           <3QL>
310 FOR Y=0 TO 8: FOR X=0 TO 8: READ FE(X,Y): NEX
    T X,Y: SLOW                                         <S30>
320 REM --- JOY - ABFRAGE ---                           <J2C>
330 J2=JOY(2): T1=PEEK(211): IF J2=0 AND T1=0 GOT
    O 330                                               <A91>
340 IF J2=128 GOTO 530                                  <R37>
350 IF J2>9 THEN 330                                    <7CN>
360 ON T(T1) GOTO 90,740,80                             <MOQ>
370 ON V(J2) GOTO 410,440,470,500                       <6BB>
380 GOTO 330                                            <RE1>
390 REM --- UNTERPROGRAMME ---                          <HQ7>
400 REM --- JOY = 1 (UP) ---                            <G20>
410 Y1=Y1-1: IF FE(X1,Y1)=2 THEN Y1=Y1+1: GOTO 33
    0                                                   <JGU>
420 GOSUB 40: GOTO 330                                  <6JF>
430 REM --- JOY = 5 (DOWN) ---                          <GQ3>
440 Y1=Y1+1: IF FE(X1,Y1)=2 THEN Y1=Y1-1: GOTO 33
    0                                                   <GBI>
450 GOSUB 40: GOTO 330                                  <6FC>
460 REM --- JOY = 3 (RIGHT) ---                         <HI6>
470 X1=X1+1: IF FE(X1,Y1)=2 THEN X1=X1-1: GOTO 33
    0                                                   <BIQ>
480 GOSUB 40: GOTO 330                                  <7RA>
490 REM --- JOY = 7 (LEFT) ---                          <IA9>
500 X1=X1-1: IF FE(X1,Y1)=2 THEN X1=X1+1: GOTO 33
    0                                                   <BAM>
510 GOSUB 40: GOTO 330                                  <6NF>
520 REM --- JOY = MITTE + FEUERKNOPF ---                <G20>
530 IF EN=1 GOTO 550                                    <AC0>
540 IF FE(X1,Y1)=0 GOTO 330: ELSE X0=X1: Y0=Y1: E
    N=1: MOVSPR 2,74+(24*X1),51+(24*Y1): SOUND 1,
    800,SO: SPRITE 2,1,4: FOR TY=1 TO 30: NEXT :
    GOTO 330                                            <GGT>
550 IF X1<>X0 AND Y1<>Y0 OR FE(X1,Y1)=1 THEN GOSU
    B 60: GOTO 330                                      <5TR>
560 IF X1<>X0 THEN XM=INT(((X1+X0)/2)+1)-1: YM=Y1
    : ELSE YM=INT(((Y1+Y0)/2)+1)-1: XM=X1               <F0V>
570 IF X0=X1 THEN 600                                   <SK2>
580 IF X0+2<>X1 AND X0-2<>X1 THEN GOSUB 60: GOTO
    330                                                 <Q07>
590 GOTO 610                                            <OQ9>
600 IF Y0+2<>Y1 AND Y0-2<>Y1 THEN GOSUB 60: GOTO
    330                                                 <F0V>
610 IF FE(XM,YM)=0 THEN GOSUB 60: GOTO 330              <TGU>
620 GOSUB 640: GOTO 330                                 <Q4P>
630 REM --- KUGELN NEU SETZEN ---                       <HQ7>
640 :                                                   <7CT>
650 MOVSPR 3,74+(24*X1),51+(24*(Y1)): MOVSPR 4,74
    +(24*XM),51+(24*YM)                                 <78T>
660 COLOR 1,14: ZA=ZA-1: ZA$=STR$(ZA): CHAR ,8,6,
    "(2SPACE)": CHAR ,7,6,(ZA$): COLOR 1,8              <ION>
670 SPRITE 3,1,8: SPRITE 4,1,1: SPRITE 2,1,1: GSH
    APE A$,50+(24*X0),25+(24*(Y0-1)),4                  <BS0>
680 SOUND 1,1000,SD: GSHAPE A$,50+(24*X1),25+(24*
    (Y1-1)),0: GSHAPE A$,50+(24*XM),25+(24*(YM-1)
    ),4                                                 <S47>
690 SPRITE 2,0: SPRITE 3,0: SPRITE 4,0: EN=0: FE(
    X0,Y0)=0: FE(X1,Y1)=1: FE(XM,YM)=0                  <B9K>
700 S1(ZA)=X1: S2(ZA)=Y1: D1(ZA)=X0: D2(ZA)=Y0: M
    1(ZA)=XM: M2(ZA)=YM                                 <BBB>
710 RETURN                                              <K27>
720 END                                                 <H66>
730 REM --- KUGEL - POS. WIEDERHERSTELLEN ---           <HA5>
740 IF ZA>31 THEN GOTO 330                              <V1V>
750 MOVSPR 2,74+(24*D1(ZA)),51+(24*D2(ZA)): MOVSP
    R 3,74+(24*S1(ZA)),51+(24*S2(ZA)): MOVSPR 4,7
    4+(24*M1(ZA)),51+(24*M2(ZA)): COLOR 1,8             <I2F>
760 SPRITE 2,1,8: SPRITE 3,1,1: SPRITE 4,1,8            <7LU>
770 ZA=ZA+1: ZA$=STR$(ZA): COLOR 1,14: CHAR ,7,6,
    (ZA$): COLOR 1,8: ZA=ZA-1                           <M7Q>
780 GSHAPE A$,50+(24*S1(ZA)),25+(24*(S2(ZA)-1)),4
    : GSHAPE A$,50+(24*D1(ZA)),25+(24*(D2(ZA)-1))
    ,0: GSHAPE A$,50+(24*M1(ZA)),25+(24*(M2(ZA)-1
    )),0                                                <D28>
790 SPRITE 2,0: SPRITE 3,0: SPRITE 4,0                  <OED>
800 FE(S1(ZA),S2(ZA))=0: FE(D1(ZA),D2(ZA))=1: FE(
    M1(ZA),M2(ZA)) =1                                   <HDV>
810 ZA=ZA+1: GOTO 330                                   <0J2>
```

Listing 1. »Solitaire« (Schluß)

CHECKSUMMER 128 & MSE

Diese beiden Programme sind unentbehrlich beim Abtippen unserer Listings. Sie helfen Tippfehler zu vermeiden und sparen eine Menge Zeit.

Nobody is perfect. Jeder Computer-Fan, egal ob blutiger Anfänger oder ausgefuchster Profi, macht beim Abtippen von Programmen Tippfehler. Diese Fehler später zu finden, kann ein langwieriges Unterfangen werden.

Deshalb haben wir für Sie die Programme »Checksummer 128« und »MSE« (Maschinen-Sprache-Editor) entwickelt. Der Checksummer ist für Basic-Programme und der MSE für Maschinensprache-Listings zuständig.

Der MSE

Wie eben erwähnt, dient der MSE zur Eingabe von Maschinensprache-Programmen. Als erstes müssen Sie den sogenannten »MSE-Lader« (Listing 1) abtippen. Dieser erzeugt das eigentliche MSE-Programm auf Diskette oder Kassette.

```
5 PRINT CHR$(14)                                          <KD2>
10 PRINT "{CLR}"                                          <1HM>
20 PRINT "Q******************************W"               <G90>
30 PRINT "{4DOWN,2SPACE}TEST{2SPACE,BLUE,6SPACE}"         <QGS>
40 PRINT "BBBBBBBBBBBBBBBBBBBBBBBBBBBBBB"                 <BAG>
```

Bild 1. Die Bedeutung der Steuerzeichen wird im nachfolgenden Text erklärt

In Zeile 10 müssen Sie nach den Anführungsstrichen die Tasten <SHIFT CLR/HOME> drücken und nicht die Klammern mit dem Wort CLR eingeben. In Zeile 20 drücken Sie nach den Anführungsstrichen die CBM-Taste und den Buchstaben <Q>, gefolgt von mehreren SHIFT- und Stern-Tasten und zum Schluß die CBM-Taste und den Buchstaben <W>. In Zeile 30 ist es viermal die CURSOR-abwärts-Taste, gefolgt von zweimaliger Leertaste, dann <SHIFT T> und normal EST, zum Schluß noch zweimal die Leertaste, die Farbtaste Blau <CTRL 7> und sechsmal die Leertaste. Zeile 40 besteht lediglich aus mehreren Grafikzeichen, die mit der CBM-Taste und <B> erzeugt werden.

Wichtig: Vor dem Eintippen des MSE-Laders müssen Sie unbedingt ein paar Befehle eingeben (ohne Basic-Zeilennummer): POKE 44,32 : POKE 8192,0 : NEW und <RETURN> drücken. Die Eingabe des MSE-Laders muß im C 64-Modus erfolgen, ebenso die Eingabe der MSE-Listings!

Jetzt können Sie beginnen, das Listing 1 abzutippen. Der MSE-Lader erkennt, wenn Sie beim Eintippen der DATA-Zeilen einen Fehler gemacht haben.

Wenn Sie das Listing 1 nicht auf einmal abtippen möchten, müssen Sie vor jedem neuen Laden des Programms unbedingt die oben genannte POKE-Zeile eingeben!

Datasetten-Besitzer müssen die »8« am Ende von Zeile 343 in eine »1« ändern.

Wenn Sie alles richtig gemacht haben und das Programm fehlerfrei abgetippt wurde, speichert es sich selbst auf Diskette oder Kassette unter dem Namen »MSE V1.1«. Dieses fertige MSE-Programm laden Sie dann bei Bedarf wie ein normales Basic-Programm und starten es mit »RUN«.

So arbeitet man mit dem MSE

Als erstes möchte der MSE den Namen des zu bearbeitenden Programms wissen. Dieser steht in der ersten Zeile unserer MSE-Listings. Dann müssen Sie die Start- und Endadresse des Programms eingeben. Dies sind die letzten beiden, vierstelligen Hexadezimalzahlen in der ersten Zeile unserer Listings.

Wenn Sie ein Programm von Diskette oder Kassette laden wollen, um an einer bestimmten Stelle weiterzutippen oder noch eine Korrektur vorzunehmen, geben Sie auf die Frage nach der Startadresse ein »L« ein. Danach müssen Sie »D« oder »T« drücken, je nachdem, ob Sie von Diskette oder Kassette (»tape«) laden möchten. Wenn das Programm unter diesem Namen nicht auf der Diskette vorhanden ist oder ein sonstiger Ladefehler vorlag, meldet sich der MSE mit »I/O-ERROR«. In so einem Fall drücken Sie <RUN/STOP RESTORE> und geben einfach noch einmal »RUN« ein.

Beim Abtippen geben Sie hintereinander die abgedruckten Buchstaben und Zahlen des jeweiligen Listings ein. Für die Leerzeichen zwischen den Zahlen sorgt der MSE automatisch selbst. Wenn Sie in einer Zeile einen Tippfehler gemacht haben, meldet sich der MSE sofort mit einem Brummton und der Meldung »EINGABEFEHLER«. Nach einem Druck auf die <RETURN>-Taste können Sie mit der DEL-Taste den Fehler korrigieren.

Wenn Sie das gewünschte Programm vollständig eingegeben haben, speichert es der MSE automatisch auf Diskette oder Kassette.

Bei längeren Listings ist es unwahrscheinlich, daß Sie das komplette Programm auf einmal eingeben. Sie können Ihre bisherige Tipparbeit jederzeit durch <CTRL S> auf Diskette oder Kassette speichern und Ihr Werk später fortsetzen. Sie sollten sich dann allerdings im Heft markieren, wie weit Sie beim Abtippen gekommen sind! Später geben Sie dann nach dem Laden des ersten Programmteils <CTRL N> ein und auf die dann folgende Frage nach der Startadresse die Zeilennummer (Adresse), bei der Sie aufgehört haben zu tippen.

<CTRL M> erlaubt Ihnen jederzeit, Ihr Werk listen zu lassen. Durch <SPACE> können Sie weiterlisten lassen und durch <RUN/STOP> das Listen abbrechen.

{CTRL}	steht für Control-Taste, so bedeutet {CTRL+A}, daß Sie die Control-Taste und die Taste »A« drücken müssen. Im folgenden steht:
{DOWN}	Taste neben rechtem Shift, Cursor unten
{UP}	Shift-Taste & Taste neben rechtem Shift; Cursor hoch
{CLR}	Shift-Taste & 2. Taste von ganz rechts oben
{INST}	Shift-Taste & Taste ganz rechts oben
{HOME}	2. Taste von ganz rechts oben
{DEL}	Taste ganz rechts oben
{RIGHT}	Taste ganz rechts unten
{LEFT}	Shift-Taste & Taste unten rechts
{SPACE}	Leertaste, Hinweis: {13 SPACE} bedeutet 13mal die Leertaste drücken
{SHIFT-SPACE}	Shift-Taste & Leertaste
{F1}	grauer Tastenblock rechts oben
{F3}	grauer Tastenblock rechts oben
{F5}	grauer Tastenblock rechts oben
{F7}	grauer Tastenblock rechts oben
{F2}	grauer Tastenblock rechts oben & Shift
{F4}	grauer Tastenblock rechts oben & Shift
{F6}	grauer Tastenblock rechts oben & Shift
{F8}	grauer Tastenblock rechts oben & Shift
{RETURN}	Return-Taste
{CTRL+I}	TAB-Taste

{CTRL+J}	Line-Feed-Taste
{CTRL}	Control-Taste
{BLACK}	Control-Taste & 1
{WHITE}	Control-Taste & 2
{RED}	Control-Taste & 3
{CYAN}	Control-Taste & 4
{PURPLE}	Control-Taste & 5
{GREEN}	Control-Taste & 6
{BLUE}	Control-Taste & 7
{YELLOW}	Control-Taste & 8
{RVSON}	Control-Taste & 9
{RVOFF}	Control-Taste & 0
{ORANGE}	Commodore-Taste & 1
{BROWN}	Commodore-Taste & 2
{LIG.RED}	Commodore-Taste & 3
{GREY 1}	Commodore-Taste & 4
{GREY 2}	Commodore-Taste & 5
{LIG.GREEN}	Commodore-Taste & 6
{LIG.BLUE}	Commodore-Taste & 7
{GREY 3}	Commodore-Taste & 8

Wenn Sie sich erst einmal an die in Klartext geschriebenen Steuerzeichen gewöhnt haben, werden Sie den Vorteil dieser Schreibweise erkennen. Der zu dem jeweiligen Steuerzeichen gehörende Klartext ist so verfaßt, daß Sie leicht die Taste beziehungsweise die Tastenkombination finden, die Sie drücken müssen.

Tabelle 1. Die Steuerbefehle für den C128 im Klartext

Wenn Sie einen Drucker besitzen, können Sie das Programm auch mit <CTRL P> ausdrucken.

Mit <CTRL L> wird das Programm noch einmal neu in Ihren C 128 geladen.

Der Checksummer 128

Zuerst einmal müssen Sie das Checksummer-Programm (siehe Listing 2) mit dem MSE im C 64-Modus abtippen. Die Eingabehinweise entnehmen Sie bitte obigen Angaben zur Eingabe von MSE-Listings. Wenn Sie fertig sind, speichern Sie das Programm auf Diskette oder Kassette.

Jetzt geht es los:

1. Starten Sie den Checksummer (im C 128-Modus) durch die Eingabe von »RUN« und das Drücken der RETURN-Taste.
2. Wenn die Meldung »Checksummer 128 aktiv « auf dem Bildschirm erscheint, haben Sie keinen Tippfehler gemacht und der Checksummer ist nun eingeschaltet.
3. Nun können Sie den Checksummer testen. Geben Sie bitte folgende Zeile ein und drücken Sie die RETURN-Taste:

10 REM

In der linken oberen Bildschirmecke sehen Sie nun die Prüfsumme der eben eingegebenen Basic-Zeile. Sie muß <A99> lauten. Dem Checksummer ist es übrigens egal, ob Sie »10 REM« oder »10REM« eintippen. Nur innerhalb von Anführungszeichen ist die richtige Anzahl an Leerzeichen wichtig. Diese Prüfsummen erscheinen (sofern Sie den Checksummer 128 eingeschaltet haben) immer dann,

```
100 REM DIESES PROGRAMM ERZEUGT DEN
110 REM MSE V1.1 AUF DISKETTE.
120 REM BESITZER EINER DATASETTE
130 REM MUESSEN DIE '8' AM ENDE VON
140 REM ZEILE 343 IN EINE '1' AENDERN!
150 REM
230 IF PEEK(44)<>32 THEN PRINT "{CLR}SIE HABEN VE
    RGESSEN, DIE POKES EINZUGE- BEN!": END
240 PRINT "{CLR}";: DIM H(75): FOR I=0 TO 9
250 H(48+I)=I: H(65+I)=I+10: NEXT : Z=1000
260 FOR I=2048 TO 3755 STEP 20: PRINT "{HOME}ICH
    LESE ZEILE:"Z
261 FOR N=0 TO 19: READ A$: IF LEN(A$)<>2 THEN 90
    0
262 IF PEEK(63)+PEEK(64)*256<>Z THEN 800
270 H=ASC(LEFT$(A$,1)): L=ASC(RIGHT$(A$,1))
280 D=H(H)*16+H(L): S=S+D: POKE I+N,D
290 NEXT : READ V: IF S<>V THEN 900
300 S=0: Z=Z+1: NEXT : R=PEEK(2111): H=PEEK(2106)
301 POKE 53280,R: POKE 53281,H: POKE 646,R: PRINT
     "{CLR}DIE DATA-ZEILEN SIND FEHLERFREI!"
302 PRINT "SIE KOENNEN NUN DIE FARBEN DES MSE"
303 PRINT "EINSTELLEN.": PRINT "{2DOWN} {RVSON}DR
    UECKEN SIE <1>, <2> ODER <9>
304 PRINT "{DOWN,2SPACE}<1> - RAHMEN-/SCHRIFTFARB
    E
305 PRINT "{2SPACE}<2> - HINTERGRUNDFARBE
306 PRINT "{DOWN,2SPACE}<9> - FARBEN UEBERNEHMEN
307 PRINT "{2DOWN}FARBE <1>:"R: PRINT "FARBE <2>:
    "H
308 GET A: IF A=0 THEN 308
309 IF A=1 THEN R=(R+1) AND 15
310 IF A=2 THEN H=(H+1) AND 15
311 IF A=9 THEN 340
312 GOTO 301
340 POKE 2106,H: POKE 2111,R
342 POKE 631,19: POKE 632,13: POKE 198,2
343 PRINT "{CLR}SAVE" CHR$(34)"MSE V1.1" CHR$(34)
    ",8
344 POKE 43,1: POKE 44,8: POKE 45,172: POKE 46,14
    : END
800 PRINT "{CLR,RVSON}SIE HABEN ZEILE"Z"{LEFT} VE
    RGESSEN:": A=PEEK(646) AND 15
810 POKE 646,PEEK(53281) AND 15: PRINT "LIST"Z-2"
    -"Z+2: POKE 646,A
820 GOTO 920
900 PRINT "{CLR,RVSON}SIE HABEN EINEN TIPPFEHLER
    GEMACHT:": A=PEEK(646) AND 15
910 POKE 646,PEEK(53281) AND 15: PRINT "LIST"Z: P
    OKE 646,A
920 POKE 631,19: POKE 632,17: POKE 633,13: POKE 1
    98,3: END
1000 DATA 00,0B,08,0A,00,9E,32,30,36,31,00,00,00,
     A2,08,A9,36,85,A4,A9, 1247
1001 DATA 08,85,A5,A9,00,85,A6,A9,B0,85,A7,A0,00,
     B1,A4,91,A6,C8,D0,F9, 2888
1002 DATA E6,A5,E6,A7,CA,D0,F2,A9,36,85,01,4C,00,
     B0,20,D1,B1,A9,00,8D, 2781
1003 DATA 21,D0,A9,0F,8D,20,D0,8D,86,02,A0,B3,A9,
     74,20,FF,B1,A0,B3,A9, 2679
1004 DATA B9,20,FF,B1,A0,00,20,CF,FF,99,01,02,C8,
     C9,0D,D0,F5,88,F0,D2, 2912
1005 DATA C0,11,90,02,A0,10,8C,00,02,20,EA,B1,A0,
     B3,A9,CF,20,FF,B1,20, 2327
1006 DATA 8E,B4,85,FC,85,62,20,8E,B4,85,FB,85,61,
     20,A7,B4,D0,20,A0,B3, 2864
1007 DATA A9,E5,20,FF,B1,20,8E,B4,85,60,20,8E,B4,
     85,5F,20,A7,B4,D0,0A, 2624
1008 DATA A5,61,C5,5F,A5,62,E5,60,90,06,20,43,B3,
     4C,3A,B0,A9,AA,A0,00, 2379
1009 DATA EA,EA,E6,FB,D0,02,E6,FC,20,3F,B2,90,EF,
     4C,FB,B4,A2,02,86,58, 3190
1010 DATA A9,A6,A0,9D,20,F2,B1,20,E4,FF,F0,FB,C9,
     30,90,0C,C9,47,B0,08, 2970
1011 DATA C9,3A,90,0B,C9,41,B0,07,C9,14,D0,0F,4C,
     0B,B1,20,D2,FF,A6,58, 2322
1012 DATA 95,F7,C6,58,D0,D2,60,AE,8D,02,F0,26,C9,
     0C,D0,03,4C,0B,B6,C9, 2685
1013 DATA 13,D0,03,4C,8B,B5,C9,0D,D0,03,4C,BA,B4,
     C9,10,D0,03,4C,68,B5, 2282
1014 DATA C9,0E,D0,06,20,5F,B4,4C,64,B1,4C,92,B0,
     A5,F9,20,02,B1,0A,0A, 2132
1015 DATA 0A,0A,85,F9,A5,F8,20,02,B1,05,F9,60,C9,
     3A,90,02,69,08,29,0F, 1950
1016 DATA 60,A6,59,E0,08,90,1F,A6,58,E0,02,B0,06,
     20,D2,FF,4C,8E,B0,C6, 2509
1017 DATA 59,A0,14,A9,92,20,F2,B1,CA,D0,FA,84,57,
     68,68,4C,8B,B1,A6,D3, 2891
1018 DATA E0,08,B0,03,4C,92,B0,20,D2,FF,A6,58,E0,
     02,90,09,C6,59,20,D2, 2468
1019 DATA FF,C6,58,D0,F9,4C,8E,B0,48,4A,4A,4A,4A,
     20,59,B1,68,29,0F,C9, 2419
1020 DATA 0A,90,02,69,06,69,30,4C,D2,FF,A2,FC,9A,
     20,D1,B1,20,48,B2,20, 2261
1021 DATA EA,B1,20,9F,B2,A5,FC,20,4E,B1,A5,FB,20,
     4E,B1,20,ED,B1,A9,3A, 2860
1022 DATA A0,20,20,F2,B1,A9,00,85,59,20,8E,B0,20,
     ED,B1,A4,59,20,EF,B0, 2530
1023 DATA 91,FB,C8,84,59,C0,08,90,EC,20,10,B2,A9,
     12,20,D2,FF,20,8E,B0, 2657
1024 DATA 20,EF,B0,C5,FF,F0,0D,20,43,B3,A9,14,A0,
     14,20,F2,B1,4C,A2,B1, 2665
1025 DATA A9,92,20,D2,FF,20,33,B2,20,E0,B2,20,3F,
     B2,90,9F,4C,8B,B5,A9, 2648
1026 DATA 93,20,D2,FF,A2,00,A9,03,9D,00,D8,9D,00,
     D9,9D,00,DA,9D,00,DB, 2476
1027 DATA E8,D0,EF,60,A9,0D,2C,A9,20,4C,D2,FF,20,
     D2,FF,98,4C,D2,FF,20, 2965
1028 DATA E4,FF,F0,FB,60,84,5D,85,5C,A0,00,B1,5C,
     F0,06,20,D2,FF,C8,D0, 3100
1029 DATA F6,60,A5,FB,85,5A,A0,00,84,5B,B1,FB,18,
     65,5A,85,5A,90,02,E6, 2606
1030 DATA 5B,06,5A,26,5B,C8,C0,08,90,EC,A5,5A,65,
     5B,85,FF,60,18,A5,FB, 2467
1031 DATA 69,08,85,FB,90,02,E6,FC,60,A5,FB,C5,5F,
     A5,FC,E5,60,60,A0,B3, 3106
1032 DATA A9,FB,20,FF,B1,A0,01,B9,00,02,20,D2,FF,
     CC,00,02,C8,90,F4,A9, 2692
1033 DATA 14,ED,00,02,AA,20,ED,B1,CA,D0,FA,A5,62,
     20,4E,B1,A5,61,20,4E, 2457
1034 DATA B1,20,ED,B1,A5,60,20,4E,B1,A5,5F,20,4E,
     B1,EA,EA,EA,EA,EA,EA, 3122
1035 DATA EA,EA,24,5E,10,01,60,A9,12,20,D2,FF,A2,
     28,20,ED,B1,CA,D0,FA, 2703
1036 DATA A9,92,4C,D2,FF,A5,D6,C9,16,B0,01,60,A9,
     A0,85,A4,A9,78,85,A6, 2945
1037 DATA A9,04,85,A5,85,A7,A2,13,A0,27,B1,A4,91,
     A6,88,10,F9,CA,F0,19, 2671
1038 DATA 18,A5,A4,69,28,85,A4,90,02,E6,A5,18,A5,
     A6,69,28,85,A6,90,E0, 2503
```

```
1039 DATA E6,A7,4C,B6,B2,A9,91,4C,D2,FF,A9,0F,8D,
     18,D4,A9,00,8D,05,D4, 2776
1040 DATA A9,F7,8D,06,D4,A9,11,8D,04,D4,A9,32,8D,
     01,D4,A9,00,8D,00,D4, 2413
1041 DATA A0,80,20,09,B3,A9,10,8D,04,D4,60,A2,FF,
     CA,D0,FD,88,D0,F8,60, 2914
1042 DATA A9,0F,8D,18,D4,A9,2D,8D,05,D4,A9,A5,8D,
     06,D4,A9,21,8D,04,D4, 2385
1043 DATA A9,07,8D,01,D4,A9,05,8D,00,D4,A0,FF,20,
     09,B3,A9,20,8D,04,D4, 2250
1044 DATA A9,00,8D,01,D4,8D,00,D4,60,38,20,F0,FF,
     8A,48,98,48,18,A0,06, 2179
1045 DATA A2,18,20,F0,FF,A0,B4,A9,0A,20,FF,B1,20,
     12,B3,20,E4,FF,F0,FB, 2931
1046 DATA A2,1D,A9,14,20,D2,FF,CA,D0,FA,68,A8,68,
     AA,18,4C,F0,FF,0D,0D, 2704
1047 DATA 0D,20,20,20,20,20,20,20,4D,41,53,43,48,
     49,4E,45,4E,53,50,52, 1144
1048 DATA 41,43,48,45,20,2D,20,45,44,49,54,4F,52,
     20,0D,0D,20,20,20,20, 1023
1049 DATA 20,20,20,20,56,4F,4E,20,4E,2E,4D,41,4E,
     4E,20,26,20,44,2E,57, 1128
1050 DATA 45,49,4E,45,43,4B,00,0D,0D,0D,20,20,20,
     50,52,4F,47,52,41,4D, 1102
1051 DATA 4D,4E,41,4D,45,20,3A,20,00,0D,0D,20,20,
     20,53,54,41,52,54,41, 1073
1052 DATA 44,52,45,53,53,45,20,3A,20,24,00,0D,0D,
     20,20,20,45,4E,44,41, 1014
1053 DATA 44,52,45,53,53,45,20,20,20,3A,20,24,00,
     92,01,01,50,52,4F,47, 1136
1054 DATA 52,41,4D,4D,20,3A,20,00,12,20,20,2A,2A,
     2A,20,46,41,4C,53,43, 1024
1055 DATA 48,45,20,45,49,4E,47,41,42,45,20,2A,2A,
     2A,20,20,92,00,0D,0D, 1058
1056 DATA 2A,2A,2A,20,45,4E,44,45,20,2A,2A,2A,00,
     13,01,20,20,12,44,92, 916
1057 DATA 49,53,4B,20,4F,44,45,52,20,12,54,92,41,
     50,45,0D,00,13,20,20, 1151
1058 DATA 49,2F,4F,20,2D,20,46,45,48,4C,45,52,00,
     20,D1,B1,20,48,B2,A0, 1606
1059 DATA B3,A9,CF,20,FF,B1,20,8E,B4,85,FC,20,8E,
     B4,85,FB,C5,61,A5,FC, 3207
1060 DATA E5,62,90,23,A5,FB,C5,5F,A5,FC,E5,60,B0,
     19,20,A7,B4,D0,14,60, 2860
1061 DATA 20,A7,B4,F0,0C,85,F9,20,A7,B4,F0,05,85,
     F8,4C,EF,B0,68,68,20, 2749
1062 DATA 43,B3,4C,5F,B4,20,CF,FF,C9,4C,D0,09,20,
     D1,B1,20,48,B2,4C,0B, 2372
1063 DATA B6,C9,0D,60,A9,00,85,5E,20,5F,B4,20,EA,
     B1,20,0D,B5,24,5E,30, 2042
1064 DATA 05,20,E4,FF,F0,FB,20,E1,FF,F0,26,20,9F,
     B2,24,5E,10,09,20,4E, 2435
1065 DATA B5,20,0D,B5,20,60,B5,20,33,B2,20,3F,B2,
     90,D7,A0,B4,A9,28,20, 2190
1066 DATA FF,B1,20,E4,FF,C9,0D,D0,F9,A9,00,85,5E,
     A5,61,85,FB,A5,62,85, 3056
1067 DATA FC,20,E0,B2,4C,64,B1,A5,FC,20,4E,B1,A5,
     FB,85,FF,20,4E,B1,A9, 3003
1068 DATA 20,A0,3A,20,F2,B1,A0,00,20,ED,B1,B1,FB,
     20,4E,B1,C8,C0,08,90, 2566
1069 DATA F3,20,ED,B1,24,5E,30,03,A9,12,2C,A9,20,
     20,D2,FF,20,10,B2,A5, 2190
1070 DATA FF,20,4E,B1,A9,92,20,D2,FF,4C,EA,B1,A9,
     FF,85,B8,85,B9,A9,04, 3073
1071 DATA 85,BA,20,C0,FF,A2,FF,4C,C9,FF,20,CC,FF,
     A9,FF,4C,C3,FF,20,5F, 3315
1072 DATA B4,A9,80,85,5E,20,4E,B5,20,48,B2,A2,24,
     A9,2D,20,D2,FF,CA,D0, 2596
1073 DATA FA,20,EA,B1,20,EA,B1,20,60,B5,4C,C1,B4,
     20,B8,B5,A6,5F,A4,60, 2812
1074 DATA A9,61,20,D8,FF,B0,0A,20,B7,FF,29,BF,D0,
     03,4C,FB,B4,A9,01,20, 2577
1075 DATA C3,FF,20,68,B6,A0,B4,A9,4F,20,FF,B1,20,
     F9,B1,4C,FB,B4,20,68, 2921
1076 DATA B6,A9,37,A0,B4,20,FF,B1,20,F9,B1,A2,08,
     C9,44,F0,06,A2,01,C9, 2717
1077 DATA 54,D0,F1,A9,01,A8,20,BA,FF,A0,00,E0,01,
     F0,1A,A9,40,8D,20,02, 2403
1078 DATA A9,3A,8D,21,02,B9,01,02,99,22,02,C8,CC,
     00,02,90,F4,C8,C8,D0, 2182
1079 DATA 0C,B9,01,02,99,20,02,C8,CC,00,02,D0,F4,
     98,A2,20,A0,02,4C,BD, 2018
1080 DATA FF,20,B8,B5,A5,BA,C9,08,90,33,A6,B9,86,
     57,A9,01,20,C3,FF,A9, 2800
1081 DATA 60,85,B9,20,C0,FF,B0,28,A5,BA,20,B4,FF,
     A5,B9,20,96,FF,20,A5, 2911
1082 DATA FF,85,61,A5,90,4A,4A,B0,13,20,A5,FF,85,
     62,20,AB,FF,A5,57,85, 2663
1083 DATA B9,A9,00,20,D5,FF,90,03,4C,A3,B5,86,5F,
     84,60,A5,BA,C9,01,D0, 2639
1084 DATA 0A,AD,3D,03,85,61,AD,3E,03,85,62,4C,FB,
     B4,A9,13,20,D2,FF,A2, 2300
1085 DATA 1C,20,ED,B1,CA,D0,FA,60,00,00,00,00,00,
     00,00,00,00,00,00,00, 1230
```

Listing 1. Der »MSE«-Lader für die Eingabe von Maschinensprache-Programmen

```
Name : checksummer 128   1c01 1e9b
----------------------------------
1c01 : 29 1c c3 07 fe 02 30 3a  3f
1c09 : 99 22 93 43 48 45 43 4b  53
1c11 : 53 55 4d 4d 45 52 20 31  d6
1c19 : 32 38 20 41 4b 54 49 56  c1
1c21 : 22 3a 9e 37 32 31 31 00  60
1c29 : 00 00 a9 80 85 9d a9 00  90
1c31 : 85 7f a9 ff 85 3c a9 59  74
1c39 : 85 4e a9 1c 85 4f a9 22  91
1c41 : 85 50 a9 0b 85 51 a0 00  1f
1c49 : a2 02 b1 4e 91 50 c8 d0  83
1c51 : f9 e6 4f e6 51 ca 10 f2  00
1c59 : 4c 2a 0b 00 00 00 00 00  7d
1c61 : a9 00 8d 00 ff a9 0d 20  2f
1c69 : d2 ff ad cf 41 c9 56 d0  fd
1c71 : 11 ad d0 41 c9 37 d0 0a  63
1c79 : ad d1 41 c9 2e d0 03 a9  61
1c81 : ff 2c a9 00 8d 25 0b aa  84
1c89 : d0 0b 8d 64 0d a9 65 85  8d
1c91 : 2b a9 0d 85 2c 20 cd 0b  96
1c99 : a9 6c 8d 02 03 a9 0b 8d  e1
1ca1 : 03 03 2c 27 0b 10 06 ee  3d
1ca9 : 27 0b 4c 35 0d 2c 26 0b  f0
1cb1 : 10 03 4c f9 0c 20 a4 0b  00
1cb9 : 20 af 0b 20 c2 0b aa f0  89
1cc1 : e1 b0 0c ce 26 0b 20 e3  da
1cc9 : 0b 20 af 0b 20 c2 0b 2c  ce
1cd1 : 25 0b 30 03 4c 94 a4 4c  7d
1cd9 : d7 4d 2c 25 0b 30 03 4c  dd
1ce1 : 60 a5 4c 93 4f a9 ff a2  21
1ce9 : 01 2c 25 0b 30 05 85 7a  e1
1cf1 : 86 7b 60 85 3d 86 3e 60  bf
1cf9 : 2c 25 0b 30 03 4c 73 00  e1
1d01 : 4c 80 03 2c 25 0b 30 03  45
1d09 : 4c 44 a6 4c d9 51 2c 25  ce
1d11 : 0b 30 03 4c 13 a6 4c 64  df
1d19 : 50 2c 25 0b 30 06 20 6b  b5
1d21 : a9 4c f1 0b 20 a0 50 a2  5c
1d29 : 00 2c 25 0b 10 02 a2 02  8a
1d31 : b5 15 85 aa b5 14 85 a9  8d
1d39 : 60 a9 00 85 a7 85 a8 a9  bb
1d41 : ff 85 fc a9 07 85 fd 20  4c
1d49 : 9d 0c f0 77 c9 20 f0 06  85
1d51 : 20 8e 0c 4c 11 0c 20 9d  72
1d59 : 0c f0 68 c9 20 f0 f7 c9  2e
1d61 : 3a d0 06 20 8e 0c 4c 20  44
1d69 : 0c c9 52 d0 29 20 9d 0c  2b
1d71 : c9 45 d0 1a 20 9d 0c c9  07
1d79 : 4d d0 0b 20 9d 0c f0 43  7a
1d81 : c9 3a d0 f7 f0 dd a5 fc  29
1d89 : d0 02 c6 fd c6 fc a5 fc  b0
1d91 : d0 02 c6 fd c6 fc a5 fc  b8
1d99 : d0 02 c6 fd c6 fc 20 9d  ec
1da1 : 0c f0 20 c9 20 f0 f7 c9  64
1da9 : 3a f0 b8 c9 22 f0 06 20  c5
1db1 : 8e 0c 4c 68 0c 20 8e 0c  7a
1db9 : 20 9d 0c f0 06 c9 22 d0  a2
1dc1 : f4 f0 ec 60 06 a7 26 a8  fc
1dc9 : 08 46 a7 28 26 a7 45 a7  e7
1dd1 : 85 a7 60 e6 fc d0 02 e6  4b
1dd9 : fd a0 00 b1 fc 60 ea ea  b1
1de1 : ea ea ea ea ea ea ea ea  e0
1de9 : ea a5 a8 45 a7 85 a8 20  02
1df1 : bf 0c 20 bf 0c a9 00 a2  0a
1df9 : 04 06 a7 26 a8 2a ca 10  d6
1e01 : f8 c9 0a 90 02 69 06 69  c9
1e09 : 30 4c d2 ff a9 00 8d 00  e5
1e11 : ff 38 20 f0 ff 8a 48 a9  1b
1e19 : 13 20 d2 ff a9 3c 20 d2  93
1e21 : ff 20 b3 0c a9 3e 20 d2  51
1e29 : ff 68 aa 18 4c f0 ff ad  b2
1e31 : 27 03 48 ad 26 03 48 a9  91
1e39 : 49 8d 26 03 a9 0d 8d 27  ba
1e41 : 03 2c 25 0b 10 03 a2 02  ad
1e49 : 2c a2 00 a5 a9 95 14 a5  5e
1e51 : aa 95 15 20 d8 0b a9 00  9c
1e59 : 8d 28 0b 2c 25 0b 30 08  be
1e61 : ce 27 0b a0 02 4c d8 a6  cd
1e69 : 20 f8 50 20 03 0c ee 26  b6
1e71 : 0b 68 8d 26 03 68 8d 27  d0
1e79 : 03 20 d6 0c 4c 6c 0b 85  23
1e81 : ab 8a 48 ae 28 0b a5 ab  22
1e89 : c9 0d d0 02 a9 00 9d 00  5e
1e91 : 08 ee 28 0b 68 aa a5 ab  46
1e99 : 18 60 ff ff ff ff ff f5  cd
```

Listing 2. Der »Checksummer 128« für die Eingabe von Basic-Listings

Impressum

Herausgeber: *Carl-Franz von Quadt, Otmar Weber*

Chefredakteur: *Georg Klinge*
Stellv. Chefredakteur: *Gottfried Knechtel – verantwortlich für den redaktionellen Teil*
Leitender Redakteur: *Klaus Schrödl*
Redaktion: *Ralf Sablowski, Elmar Friebe, Klaus Sonnenleiter*
Redaktionsassistenz: *Andrea Kaltenhauser, Brigitte Bobenstetter, Sylvia Sailer Helga Weber (202)*
Hotline: *Monika Welzel (640)*
Mitarbeiter der Redaktion: *Heimo Ponnath, Nikolaus Huber, Dr. Rudolf Egg*

Alle Artikel sind mit dem Kennzeichen des Redakteurs (kn = Gottfried Knechtel, sk = Klaus Schrödl, rs = Ralf Sablowski, ef = Elmar Friebe, so = Klaus Sonnenleiter) und/oder mit dem Namen des Autors/Mitarbeiters gekennzeichnet

Art-director: *Friedemann Porscha*
Layout: *Erich Schulze (Cheflayouter), Marian Schwarz*
Fotografie: *Sabine Tennstaedt, Diddi Zill, Janos Feitser*
Titelgestaltung: *Friedemann Porscha,*
Spritzgrafik: *Ewald Standke*
Computergrafik: *Werner Nienstedt*

Auslandsrepräsentation:
Schweiz: Markt&Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042-415656, Telex: 862329 mut ch
USA: M&T Publishing Inc.; 501 Galveston Drive Redwood City, CA 94063, Telefon: (415) 366-3600, Telex 752-351
Österreich: Markt&Technik Ges. mbH Hermann Raniger, Große Neugasse 28, A 1040-Wien, Tel. 0043-222-8579455, Telex: 047-132532

Manuskripteinsendungen: *Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten worden sein, muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt&Technik Verlag AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Mit der Einsendung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt&Technik Verlag AG verlegten Publikationen und dazu, daß Markt&Technik Verlag AG Geräte und Bauteile nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben läßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.*

Produktionsleiter: *Klaus Buck (180)*

Anzeigenverkaufsleitung: *»Populäre Computerzeitschriften«: Alexander Narings (780)*
Anzeigenleitung: *Phillip Schiede (399) – verantwortlich für Anzeigen*

Anzeigenformate: *1/1 Seite ist 266 Millimeter hoch und 185 Millimeter breit (2 Spalten à 86 Millimeter oder 4 Spalten à 43 Millimeter). Vollformat 297 x 210 Millimeter.*

Anzeigenpreise: *Es gilt die Anzeigenpreisliste vom 5. Januar 1988. 1/1-Seite sw: DM 5400,-. Farbzuschlag: erste und zweite Farbe aus der Europa-Skala je DM 1000,-. Vierfarbzuschlag DM 2800,-. Plazierung innerhalb der redaktionellen Beiträge. Mindestgröße 1/4-Seite.*

Anzeigenverwaltung und Disposition: *Lisa Landthaler (233)*

Anzeigen-Auslandsvertretung: England: *F. A. Smyth&Associates Limited, 23a, Aylmer Parade, London, N2 OPQ. Telefon:0044/1/3405058, Telefax: 0044/1/3419602*
Taiwan: *Third Wave Publishing Corp., 1-4 Fl. 977 Min Shen E. Road, Taipei 10581, Taiwan, R.O.C., Tel. 00886/2/7630052, Telefax: 00886/2/7658767, Telex: 078529335*

Vertriebsleiter: *Helmut Grünfeldt (189)*

Leiter Vertriebs-Marketing: *Benno Gaab (740)*

Vertrieb Handelsauflage: *Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebs GmbH, Hauptstätter Straße 96, 7000 Stuttgart 1,*

Bezugsmöglichkeiten: *Leser-Service: Telefon (089) 4613-249. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen.*

Preis: *Das Einzelheft kostet DM 14,–*

Druck: *SOV Graphische Betriebe, Laubanger 23, 8600 Bamberg*

Redaktionsdirektor: *Michael M. Pauly*

Vorstand: *Otmar Weber (Vors.), Bernd Balzer, Werner Brodt*

Leiter Unternehmensbereich »Populäre Computerzeitschriften«:
Eduard Heilmayr, Werner Pest

Redaktionskoordination »Populäre Computerzeitschriften«:
Hans-Günther Beer

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen: *Markt&Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0, Telex 5-22052*

ISSN 0931–8933

Telefon-Durchwahl im Verlag: *Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 089/4613 und dann die Nummer, die in den Klammern hinter dem jeweiligen Namen angegeben ist.*

wenn Sie eine Basic-Zeile eintippen und dann die RETURN-Taste drücken. In diesem Sonderheft finden Sie die Prüfsumme immer am Ende jeder Programmzeile.

Diese Zahlen dürfen Sie NICHT mit abtippen. Sie dienen lediglich zur Kontrolle, ob Sie alles richtig eingegeben haben.

Als Beispiel können Sie Bild 1 betrachten. Am rechten Rand jeder Spalte sehen Sie die Prüfsummen in spitzen Klammern.

Damit sind wir beim zweiten wichtigen Punkt: Sehen Sie sich die Zeile 343 von Listing 1 genauer an. Nach dem ersten Anführungszeichen nach dem PRINT-Befehl sehen Sie ein Zeichen, das Sie auf der Tastatur des C 128 vergeblich suchen werden: die geschweifte Klammer »{«. Immer, wenn Sie in einem unserer Listings diese Klammern sehen, dürfen Sie das, was innerhalb der Klammern steht, nicht eintippen. Sie müssen die entsprechende Taste drücken.
Beispiel: 10 PRINT"{CLR}"
bedeutet: Nach dem Anführungszeichen die »Bildschirmlöschen«-Taste drücken (<SHIFT CLR/HOME>). In Tabelle 1 sehen Sie eine Zusammenfassung aller möglichen Steuertasten und des entsprechenden Klartextes.

Wenn Sie Bild 1 betrachten, sehen Sie in Zeile 30 innerhalb der Anführungszeichen ein unterstrichenes »T« nach der geschweiften Klammer. Das bedeutet, daß Sie ein »T« zusammen mit der SHIFT-Taste drücken müssen, also <SHIFT T>. Wenn ein Zeichen »überstrichen« ist, müssen Sie dieses zusammen mit der CBM-Taste eingeben. Die CBM-Taste befindet sich ganz links unten auf der Tastatur und hat die Aufschrift »C=«. Auf dem Bildschirm sehen Sie die entsprechenden Grafikzeichen (siehe Handbuch zum Computer, im Anhang).

Synthetische Steuerzeichen

Bitte beachten Sie, daß beim C128 mit größerer Wahrscheinlichkeit als beim C64 synthetische Steuerzeichen auftreten können (SYNTH.:XXX). Dies ergibt sich aus der Menge an zusätzlichen (Bildschirm-) Steuerzeichen, besonders für den 80-Zeichen-Modus (Blinken ein, Blinken aus, Unterstreichen ein, Unterstreichen aus...), die sich nicht alle direkt über die Tastatur eingeben lassen. Ist erkennbar, daß die Basic-Zeile fehlerfrei abgetippt ist, Sie also auf die Prüfsumme verzichten können, so erscheint die korrekte Eingabe sehr leicht. Geben Sie anstelle des »SYNTH.:XXX« folgende Anweisungen ein:

```
"+CHR$(XXX)+"
```

Die CHR$-Anweisung ist dabei nicht innerhalb der Anführungsstriche, sondern außerhalb!

Wollen Sie auf die Prüfsumme nicht verzichten, ist die Eingabe etwas komplexer. Arbeiten Sie dazu folgende sieben Schritte ab:

- Subtrahieren Sie von der Zahl nach der Angabe »SYNTH.:« den Wert 128.
- Suchen Sie aus der Zeichencode-Tabelle im Handbuch (Anhang A-5) den dazugehörigen Buchstaben aus Satz 1 aus.
- Geben Sie die Basic-Zeile wie gewohnt ein und berücksichtigen das synthetische Steuerzeichen **nicht**, lassen aber einen Freiraum für ein Zeichen.
- Schließen Sie die Eingabe mit <RETURN> ab.
- Fahren Sie mit dem Cursor auf den Freiraum in der Eingabezeile.
- Schalten Sie mit <CTRL 9> den Revers-Modus ein.
- Geben Sie den Buchstaben, den Sie vorher aus Anhang A-5 ermittelt haben, geSHIFTet ein und drücken Sie <RETURN>. Die Prüfsumme müßte nun stimmen.

(G. Möllmann, N. Mann, D. Weineck, A. Lietz/kn)

Faszinierende Spielewelt

Laß' Dich in eine abenteuerliche Spielewelt entführen:

**alles, was Du dazu brauchst, ist ein C64 oder ein C128, die Spieledisketten – und schon kann die Reise losgehen!
Nutze Deine Joystick-Künste, indem Du sicher einen Weg aus dem Labyrinth findest. Bewahre Deinen kühlen Kopf in aufregenden Aktionszenen und Kampfduellen! Zeige Deine Fähigkeiten als Börsenmakler in lebensnahen Wirtschaftssimulationen! Beweise Deine Fähigkeiten als Präsident, Manager und Trainer eines Bundesligaclubs und und und...**



Die 64'er-Spielesammlung, Band 1, 1987, 115 Seiten, inklusive Diskette



Mit den 15 spannenden Spielen, der ausführlichen Anleitung sowie den farbigen Bildschirmfotos ist Dir ein fantastisches Spielvergnügen gewiß:
Balliard: Einfallswinkel = Ausfallswinkel. Wer das nicht befolgt, hat es schwer bei dieser Mischung aus Tennis und Billard. **The Way:** Zu verschlungenen Pfaden gesellen sich Geldsäcke und böse Geister, die es zu bekämpfen gilt. **Vager 3**: Joystickprofis mit ungetrübtem Visierblick und Trefferinstinkt können ihr Punktekonto schwer mit Abschußprämien beladen. **Firebug**: Hoffentlich fängt Dein Joystick nicht ebenfalls Feuer, wenn es heißt, die wertvollen Koffer aus dem brennenden Haus des Professors zu erwischen. **Pirat**: Taktik, Timing und gute Navigationskenntnisse sind Voraussetzung für ein bis zu 25 Jahre langes Piratenleben. **Wirtschaftsmanager**: Simulation aus den höchsten Etagen der Wirtschaft, nicht 1000 Stück, sondern ganze Firmen gehen über den »Ladentisch«. **Vier gewinnt**: Einfach, aber gerade deshalb ein Spiel, das schnell zu Erfolgserlebnissen führt. **Brainstorm**: Mastermind stand Pate für dieses vielseitige Denkspiel. **Hypra-Chess**: Spiel' Schach gegen einen C64. **Maze:** Wer die Übersicht behält und nicht kopflos herumspringt, wird das Ziel erreichen. **Schiffe versenken:** Endlich eine faire Version dieses weitverbreiteten Spieles, mit zusätzlichen Spielvarianten. **Handel:** Hier kannst Du deinen Geschäftssinn und Deine Risikobereitschaft unter Beweis stellen, ohne wirklich später am Hungertuch nagen zu müssen. **Börse:** Diese Spiel bietet wirklich einen hervorragenden Einstieg ins Börsenkarussell. Außerdem sind noch die Spiele **Vier in vier** und **Magic-Cubs** enthalten.

Hardware-Anforderungen:
C64 oder C128 bzw. C128D (64er-Modus), Floppy 1541, 1570 oder 1571 und Joystick.

Bestell-Nr. 90429, ISBN 3-89090-429-7

DM 39,–* (sFr 35,90*/öS 331,90*)

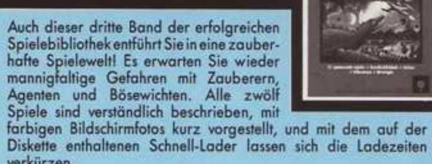
Die 64'er-Spielesammlung, Band 2, 1987, 98 Seiten, inklusive Diskette



Billard: Banden- oder Lochbillard, allein oder zu zweit, das ist hier die eine Frage. Wie Du die Kugeln dann einlochst, eine ganz andere. **Tonti:** Ein schnelles Auge und geschickte Hände am Joystick werden verlangt, um auch einmal eine Tontaube zu treffen. **Freiheit:** Du bist auf einer Insel, gefangen im untersten Verlies. Wie flüchtest Du zur Nachbarinsel? Der Weg wird schwierig, denn bis zu hundert Räume sind zu meistern. Ein Textadventure besonderer Klasse. **Apokalypse Now:** Ein Hubschrauber-Klassiker, ein Spiel mit tödlichem Ausgang – nicht unbedingt für Dich – aber nur, wenn Du am Joystick Profiqualitäten vorweisen kannst. **Black out:** Ein Spielhallenhit für den C64. Reaktionsvermögen, eine ruhige Hand und ein bißchen Glück gehören schon dazu, um alle Blöcke abzuschießen. **Aquantor:** Wasser, Wasser, Wasser, so weit der Bildschirm reicht. Nur schnell einen Damm errichten und fleißig punkten, wenn man im Trockenen sitzt. **Tödliches Dioxin**: Du hast vier hochmoderne Hubschrauber und bist selbst ein wagemutiger, unerschrockener Pilot. Der Haken an der Sache ist die Fracht – tödliches Dioxin. **Libra:** Du fliegst für die intergalaktische Föderation der Raumritter und rettest ein unabhängiges Sonnensystem. **Dasher:** Pac-Man stand Pate für dieses Spiel. Zu den bekannten Funktionen, wie Punkte fressen und Gegner vernichten, erhältst Du die Möglichkeit, eigene Spielfelder zu entwerfen und aufzubauen. **Bundesliga Manager:** Einer der 18 Bundesligaclubs wartet auf Dein Multitalent und Du hast wirklich mächtige Mittel zur Hand, um erfolgreich zu agieren. Außerdem sind noch die Spiele **Golf, Zauberschloß, Steel Slab** und **Space Invader** enthalten.

Bestell-Nr. 90428, ISBN 3-89090-428-9

DM 39,–* (sFr 35,90*/öS 331,90*)



Die 64'er-Spielesammlung, Band 3, 1988, 103 Seiten, inklusive Diskette

Auch dieser dritte Band der erfolgreichen Spielebibliothek entführt Sie in eine zauberhafte Spielewelt! Es erwarten Sie wieder mannigfaltige Gefahren mit Zauberern, Agenten und Bösewichten. Alle zwölf Spiele sind verständlich beschrieben, mit farbigen Bildschirmfotos kurz vorgestellt, und mit dem auf der Diskette enthaltenen Schnell-Lader lassen sich die Ladezeiten verkürzen.
Arabian Treasurehunt: Im Morgenland einen magischen Teppich zu finden, dürfte keine leichte Aufgabe sein, oder? **Block'n Bubble:** Wenn Sie sich schon als Zauberer betätigen mußten, warum dann so, daß Sie von Ihren eigenen Produkten gefährdet werden? **Robo's Reenge:** Helfen Sie dem armen Robo-Egg durch ein gefährliches Labyrinth! **Race of the Bones:** Hier sind Nerven gefragt. Es ist schließlich nicht alltäglich, daß man Knochen sammeln geht... **Quadranoid:** Eine äußerst professionelle Umsetzung des bekannten Spielhallenhits Arcanoid. **Future Race:** Zwei Personen können auf je einem halben Bildschirm Rennen austragen. Sie glauben das nicht? Schauen Sie rein! **Risiko:** Die Politiker können das nicht. Da müssen in der Kriegsführung so erfahrene Leute wie Sie ran. **Copter-Fight:** Hindern Sie einen Spion am Verlassen der Stadt. **Asteroids 64:** Die erste Umsetzung des bekannten Automatenspiels. **Odyssey:** Hier gilt es, von einer vermeintlich einsamen Insel wieder in die Zivilisation zurückzufinden. **Verminator:** Sie besitzen einen Kampfpanzer der neuesten Art und müssen eine ganze Stadt vor dem Untergang retten. **Der kleine Hobbit:** Diese deutsche Version des bekannten Adventures hält sich mehr an das Buch als das englische Original!

Hardware-Anforderungen
C64 oder C128 bzw. C128D (64'er Modus), Floppy 1541, 1570 oder 1571 und Joystick.

Bestell-Nr. 90596, ISBN 3-89090-596-X

DM 39,–* (sFr 35,90*/öS 331,90*)

* Unverbindliche Preisempfehlung



Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computerfachgeschäften oder in den Fachabteilungen der Warenhäuser.

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0

Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656. ÖSTERREICH: Markt&Technik Verlag Gesellschaft m.b.H., Große Neugasse 28, A-1040 Wien, Telefon (0222) 5871393-0; Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526; Ueberreuter Media Verlagsges.mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0.

125/806